Exploring the Use of Wikis for Information Sharing in Interdisciplinary Design

Jiranida Phuwanartnurak

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2013

Reading Committee:

David Hendry, Chair

Raya Fidel

Karen Fisher

Robert Mason

Program Authorized to Offer Degree:

Information School

University of Washington

Abstract

Exploring the Use of Wikis for Information Sharing in Interdisciplinary Design

Jiranida Phuwanartnurak

Chair of the Supervisory Committee:

Associate Professor David Hendry

Information School

Interdisciplinary design presents challenges in design collaboration due to the difficulty in communicating and coordinating among disciplines. Many tools have been developed and used to support information sharing in design, and the use of Web technology is becoming increasingly important for the sharing of information within design teams. Wikis have been claimed to support collaboration and information sharing. To assess this claim, the following questions could be addressed: (1) Do wikis really support design collaboration? (2) To what extent do wikis support information sharing in interdisciplinary design? And (3) Do wikis really encourage more sharing among design team members? In this dissertation I investigate these questions through an exploratory empirical study and examine how wikis might be used to support information sharing in interdisciplinary design.

Specifically, the research objectives of this dissertation are to: (1) Study how interdisciplinary design teams use wikis; (2) Identify best practices for the effective use of wikis; and (3) Examine

how wikis might be enhanced to better support information sharing in design projects. To achieve these research objectives, I studied five software design teams from three organizations that used wikis as their primary information sharing and management system. Drawing on two conceptual frameworks, the 3-T framework and Cognitive Work Analysis (CWA), I collected qualitative and quantitative data, using participant observation, semi-structured interviews, document review, and wiki review.

This dissertation enhances our understanding of how members of design teams share information among disciplines and how they use wikis to support such sharing. This dissertation uncovers and explores the conceptual issues of information sharing including information collection, information-sharing needs, and information-sharing processes. The findings reveal that the software design teams used wikis as their central information-sharing tool, largely because it seemed to provide a mechanism to link heterogeneous information from various sources into one place. The key findings of this dissertation research are as follows: (1) Wikis were used primarily to support transporting and transferring, that is, giving access to and delivering content of information items; (2) Wikis had the potential to support translating and transforming, that is, discussing and handing off information items, but teams appeared to have difficulty utilizing features that would enable these forms of information sharing; (3) Software design teams did not fully utilize the wiki's key features such as adding category tags to wiki pages, which makes it easy to browse related wiki documents, and using a discussion page associated with each individual wiki page to record related discussion; and (4) The use of wikis appeared to be influenced by many social and technical factors such as organizational environments, the nature of the design project, project team members' experiences and attitudes, and the specific kinds of wiki platforms that were used. Based on these findings, I discuss new requirements for how wikis might be designed to better support information sharing among disciplines during software design.

These dissertation findings also provide practical implications, first, for organizations and project teams that would like to adopt wiki technology to support software design and development projects and, second, for practitioners who design or develop wiki technology. Expanding on the dissertation's focus on wiki technology, its findings provide broader implications, particularly for interdisciplinary design collaboration, distributed design collaboration, and agile development. Finally, the dissertation findings contribute to the body of knowledge in the fields of information behavior, knowledge management, wiki research, and design research.

In memory of my mother,

Wanida Phuwanartnurak

For many years, you kept telling me… "I know you can do it."

Today, I wish I could tell you in person… "Mom, I did it."

And I wish I could hear you say… "I'm so proud of you."

ACKNOWLEDGEMENTS

The PhD journey has been long and challenging. Completing this journey not only requires a tremendous amount of inspiration, dedication, and self-discipline, but it also requires the support, guidance, and generosity of so many people. I would like to acknowledge some of the many people who provided me with the support and guidance necessary to complete my PhD journey.

I would like to express my sincere thanks to my committee chair, Dave Hendry, for all of his advice, guidance, and continuous support throughout my academic years at the UW iSchool. I appreciate his confidence in my abilities, his encouragement, and his kindness along the way. I could not have asked for a better advisor. I would like to extend my sincere thanks to my committee members, Raya Fidel, Karen Fisher, Roberts Mason, and Jennifer Turns, for their intellectual contributions and support throughout the dissertation process. I would also like to specifically recognize a number of faculty members from the many who have influenced me: Harry Bruce, William Jones, David McDonald, Karen Makey (UM SI), and Diane Sonnenwald (UCD SILS).

I would like to thank my iPhd fellows, especially the black cohort, for their friendship and supportive words. I would also like to thank the Academic and Collaborative Application group, UW-IT, for providing me with an opportunity to grow as a UX professional and for providing me funding through a graduate assistantship.

Most importantly, I would like to thank my family: my parents, my brother, my husband, and my son, for their unconditional love, unwavering encouragement, and support I needed to get through this long journey. To Mom and Dad, thank you for your loving support in all aspects of my life. To my brother, thank you for taking a great care of our parents while I am so far away from home pursuing a PhD. To my husband, thank you for your understanding and for taking care of me and our son. To my son, thank you for bringing me joy and keeping me sane during the dissertation process. I would also like to thank the Niedner family and the Crumpacker family for being there for me when I need a family in the States.

There are many more people that have supported me during the dissertation process in one way or another; however I cannot possibly list all of their names here. If you happen to read this dissertation, you know who you are. Thank you!

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

This chapter introduces the readers to the dissertation. First, I present the research background and the problem statement. Then, I provide a brief overview of the dissertation including research objectives and research questions. I also discuss wiki-related terminology to familiarize the readers with wiki-related terms used throughout the dissertation. Finally, I describe how the dissertation is structured.

1.1 Background

Successful design in many domains (e.g., software design, user interface design, product design, and engineering design) requires collaboration among various disciplines because, in general, individual members from a single discipline do not have all of the information, knowledge, and expertise needed for solving a design problem. As a result of the complexity of design problems, design seems to have become even more interdisciplinary than in the past (Eckert & Clarkson, 2005; Grudin & Poltrock, 1989).

Collaboration in interdisciplinary teams – consisting, for example, of a project manager, engineers, user-interface designers, researchers of various kinds, and marketing representatives – is always challenging (Bucciarelli, 1994; Curtis et al., 1988; Geisler & Rogers, 2000). In a design project, team members create and share a large amount of information, drawing on their own disciplines and from other disciplines (Poltrock et al., 2003; Sonnenwald & Lievrouw, 1997). In addition, team members from different disciplines may view and solve the same problem from different perspectives, using their own unique methods and languages, which may create barriers to information sharing. It can, therefore, be difficult for members of an interdisciplinary team to share what they know with each other. Information and communication technologies (ICTs), when configured for disciplinary and organizational constructs, can improve collaboration (Eckert et al., 2005).

Design teams employ many different kinds of Information and Communication Technologies (ICTs) to support communication and collaboration. Some design teams may use an electronic shared workspace to store design documents, to discuss shared ideas, to share information, and to collaboratively construct design artifacts. Others may use chat, instant messaging, mailing lists, and other ICTs to exchange information, to coordinate work, and to discuss work (Eckert et al., 2005). In some settings, specialized tools are often used. For example, engineering design teams may employ computer-aided design (CAD) systems to create graphical design representations on a computer system (Henderson, 1991; Perry & Sanderson, 1998). More recently, design teams have begun to employ wikis, editable websites, to do such things as: collaborating across locations; brainstorming and tracking projects; organizing fragmented knowledge and facilitating information exchange within corporations (Goodnoe, 2005; Majchrzak et al., 2006). As work is distributed across sites nationally and internationally, ICTs have assumed greater importance.

Wikis, in particular, have gained popularity as attractive collaboration tools. Wikis could change the way people work, communicate, and manage information. Indeed, perhaps wikis and similar tools have already changed how collaborative teams function (Munson, 2008). They have created new forms of collaboration and information sharing on the Web. Because the technology is relatively new, the social norms for the "proper" uses of wikis are developing, rapidly across many different types of organization, both large-scale and small. The main point of a wiki is to make web editing radically simple so that users are encouraged to add more information and to keep it current. Wikis have been claimed to provide a number of potential advantages. Most notably they are claimed to support design collaboration (Bean & Hott, 2005).

1.2 Problem Statement

> *"Not everyone needs a wiki. Not everyone wants a wiki. Not every situation benefits from becoming an open discussion or collaboration forum." – The Wiki Way (2001, p. 30).*

Many researchers have investigated the use of wikis to support distributed collaborative writing, particularly Wikipedia (e.g., Kitture & Kraut, 2010; Priolli et al., 2009; Wei et al., 2005) and teaching and learning (e.g., Ruger, 2003; Kepp & Schorr, 2009), while few studies have examined the use of wikis in design settings (Aguiar & David, 2005; Louridas, 2006; Munson, 2008; Radziwill & Shelton, 2004). Do wikis really support design collaboration? How do design teams, and individual team members, use wikis? To what extent do wikis support information sharing in interdisciplinary design? What general lessons about information sharing in design can be derived from an in-depth analysis of how teams currently employ them? This dissertation proposes to study how wikis support information sharing in interdisciplinary design.

This is an important topic because of the recent practitioner interest in wikis and also because wikis offer an interesting paradigm for studying the conceptual issues of information collection in dynamic environments, such as interdisciplinary design, where the creation and use of information is tightly coupled. It is important to note that while this dissertation proposes to investigate wikis in depth, the impact of this work is broader. Wiki technology provides a focal point for investigating more general issues such as information sharing, patterns of collaboration, and the requirements of information management tools that are effective in design.

1.3 Research Overview

The objectives of this dissertation are to: (1) Study how interdisciplinary design teams use wikis; (2) Identify best practices for the effective use of wikis; and (3) Examine how wikis might be enhanced to better support information sharing in design projects. This dissertation aims to answer the following research questions:

1. How do design teams share design-relevant information across disciplinary boundaries?
2. To what extent do wikis support information sharing and interdisciplinary design activities?
3. How could wikis be enhanced and utilized to better support information sharing in interdisciplinary design?

To answer the above research questions, I studied five software design teams from three organizations. I collected both qualitative and quantitative data, using a multi-method data collection strategy to provide rich data about the use of wikis by software design teams. These include participant observation, semi-structured interviews, document review, and wiki review (i.e. reviewing wiki site structure, content, and log file).

1.4 Terminology

The focus of this dissertation is around the use of wikis, thus wiki-related terms are used throughout the dissertation. These wiki-related terms could be confusing to those who are not familiar with wikis. To clarify what I refer to, I define wiki-related terms used in this dissertation, as follows:

Wiki is a broad term that refers to an editable website that users can easily create or edit any interlinked web pages via a web browser using a simple markup language or a rich text editor.

Wiki platform is an application that is used to run and develop wiki websites. A wiki platform is sometimes called wiki software or a wiki engine. Examples of wiki platforms include MediaWiki (http://www.mediawiki.org/wiki/MediaWiki),

Wiki site is an entire collection of wiki pages, which are interconnected by hyperlinks. In this dissertation I use the term **"project wiki site"** to denote a collection of wiki pages related to a particular project.

Wiki page refers to a single page in a wiki site.

Wiki log file is a log file automatically created and maintained by the wiki software. The wiki log file contains data about activities that occurred on wiki sites such as page view, edit history, and wiki page size.

1.5 Dissertation Structure

This dissertation is organized into seven chapters. This chapter, Chapter 1, provides an introduction to the research study, including background, problem statement, and research overview. Wiki-related terminology is also presented for readers who are not familiar with wikis. Chapter 2 presents the review of related literature from three areas: interdisciplinary design, information sharing, and wiki technology. Chapter 3 describes two conceptual frameworks and their application in the study. Chapter 4 describes the research methodology that was employed in the study, including research questions, research sites, data collection, and data analysis. It also discusses credibility and validity of this study. Chapter 5 discusses the context of wiki use, while Chapter 6 presents findings on how software design teams used wikis to share information. Chapter 7 discusses the findings, implications and contributions. Limitations of the study and recommendations for future research are also presented. Note that pseudonyms are used throughout the dissertation to protect confidentiality of individuals and the organization.

CHAPTER 2: LITERATURE REVIEW

This dissertation draws upon research from three areas: interdisciplinary design, information sharing, and wikis. Thus, this literature review is organized into three sections: interdisciplinary design; information sharing; and wiki technology. In the interdisciplinary design section, I present a discussion on design and the nature of interdisciplinary design. Then I review studies on interdisciplinary design collaboration with the aim of identifying challenges faced by interdisciplinary design teams. The review reveals that interdisciplinary design teams often have difficulty communicating and sharing information across disciplinary boundaries.

In the information sharing section, I present the definitions of information, knowledge, and information sharing, which are used in this dissertation. I also review studies on information sharing in design, and conclude that previous studies fail to adequately address multiple aspects of information processes, which are sharing information artifacts, sharing meaning, and sharing knowledge. Finally, in the wiki research section, I provide the overview of wiki. Then I review the literature relevant to wikis and their use to reflect the current state of knowledge about wikis, focusing on project wikis research. I conclude with the research gaps extracted from the wiki literature review.

2.1 Interdisciplinary Design

2.1.1 Design overview

The meaning of design is ambiguous. Design may be described in terms of the process that is followed to create something (e.g., an artifact, policy, procedure, and organizational structure), or the specific outcome of design process. In this dissertation, the focus is on "design as a process," that is, the work that is performed by multiple actors to create a successful outcome. Various definitions have been given to the term *design*, for example: "design is the initiation of change in man-made things (Jones, 1992, p. 4);" "design is a process of moving from the particular, general and universal to the ultimate particular – the specific design (Nelson & Stolterman, 2003, p. 33);" and "to design is to create something new that has not yet existed (Löwgren & Stolterman, 2004)." In the broadest sense, design refers to any activity that leads to a man-made change. Thus, all designers, regardless of their field, have a similar goal; they are aiming to add to, or to change, the real world. By definition, every design situation is unique: if the to-be-designed thing already existed, it would not need to be designed. A product of design can be abstract or concrete, big or small, important or unimportant (Löwgren & Stolterman, 2004).

Design is generally characterized as a kind of a problem-solving activity, in which the problem to be solved is a wicked problem. The term *wicked problem* was originally proposed by Rittel and Webber (1973) to describe the nature of ill-defined design and planning problems in contrast to well-defined problems that have definite solutions. Since then, many other researchers have drawn upon this concept to describe design problems (e.g., Arias et al., 2000; Cross, 1984; Goel & Pirolli, 1992; Restrepo & Christiaans, 2004). A design problem is a wicked problem in a sense that it requires an iterative process in which the need or problem is understood as the solution is generated and evaluated (Clarkson & Eckert, 2005). There is no one absolute solution to the problem: a design problem always has many acceptable solutions. Furthermore, there are no definite criteria to test a proposed design solution.

Design can be an individual problem-solving activity (Goel & Pirolli, 1992) or a social process (Bucciarelli, 1994). Because many design problems are too complex to be solved by a single individual, design becomes a collaborative process in which multiple people are involved. Thus, I argue that design should be viewed as a social process. Moreover, the increasing complexity of design calls for collaboration among various disciplines because knowledge from one discipline is not sufficient to produce a desired design outcome. For example, a software design team can consist of a project manager, engineers, user-interface designers, researchers, and marketing representatives. Various disciplines engaged in the design process have different focuses and, more often, different ways of working, thinking, and talking about design. This results in interdisciplinary design.

2.1.2 Nature of interdisciplinary design

Interdisciplinary design is sometimes referred to as "multidisciplinary design" or "cross-functional design." Drawing on the concept of communities of practices by Nardi (1993), Fischer (2001) developed the term, "communities of interest," to refer to heterogeneous design teams that bring together stakeholders from different disciplines to solve a particular design problem. An example of communities of interest is a software development team that consists of software designers, marketing specialists, psychologists, and programmers. Interdisciplinary collaboration has also been studied in the scientific research context. Palmer's (2001) study of interdisciplinary research process investigated how scientists work together to synthesize the research of two or more disciplines. I prefer the term "interdisciplinary design" to others because it emphasizes a deeper collaboration among different disciplinary groups of people. By collaboration, I mean the process, in which two or more people are working together, physically and intellectually, toward a common goal. To illustrate the nature and application of interdisciplinary design teams, I discuss two examples of interdisciplinary design effort: concurrent engineering and cross-functional product development.

Concurrent engineering

The concept of concurrent engineering was proposed as a way to reduce cost and time in product development from the traditional engineering process – sequential engineering (Prasad, 1996; Syan & Menon, 1994). In the sequential engineering process, each stage of product development, following the completion of the previous stage, is usually carried out in isolation from another. Thus, communication pathways are simplified. In contrast, concurrent engineering requires richer and more complex communication between many disciplines that are simultaneously working together. A widely accepted definition of concurrent engineering is "a systemic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from concept through disposal, including quality, cost, schedule, and user requirements" (Prasad, 1996, p. 7).

Concurrent engineering requires an interdisciplinary team approach; it emphasizes teamwork with close integration between departments (Love & Gunasekaran, 1997; Syan & Menon, 1994). At the minimum, a concurrent engineering team should consist of individuals from the following departments: product development, manufacturing engineering, marketing sales, purchasing, finance, main suppliers and customers (Syan & Menon, 1994). Therefore, concurrent engineering is an exemplar of interdisciplinary design because the process requires extensive coordination amongst people from different disciplines throughout the process. Concurrent engineering is the integration of all people, tools and resources, and information needed for product development. Compared to sequential engineering approaches, concurrent engineering approaches place greater demands on cross-disciplinary communication and information sharing.

Cross-functional product development

Organizations employ cross-functional teams for the majority of their technology and new product development. Cross-functional collaboration has been long studied in the field of management and organizational science. The use of cross-functional teams has been claimed to increase the effectiveness of product development (McDonough, 2000; Parker, 2003; Pinto et al., 1993). A cross-functional team, sometimes called a "multidisciplinary team," is composed of members, with different functional experiences and abilities, who come from diverse functional groups working together to achieve common goals. Cross-functional teams are often used in product research, design, and development to ensure that perspectives from all key functional groups are taken into account from the early design stage.

Collaboration is essential, yet challenging, for cross-functional teams because project members from different departments may view and solve the same problem from different perspectives, with their own unique method and language, all of which create barriers to information sharing (Bechky, 2003).

Characteristics of an interdisciplinary design team

In sum, interdisciplinary design is applied when a design problem is so complex that knowledge from any individual discipline is insufficient to resolve. These two examples of interdisciplinary design efforts, concurrent engineering and cross-functional product development, can be used to draw out the characteristics that describe an interdisciplinary design team. These characteristics are:

1) An interdisciplinary design team is composed of members from various disciplinary groups or functional departments;

2) Members have a shared interest in structuring and solving a design problem;

3) Members come from different groups and work closely together throughout the design process in order to solve the design problem; there are dependencies between their individual work processes and outputs.

2.1.3 Interdisciplinary design collaboration

Design collaboration has been studied from various perspectives such as systematic design methodology and the nature of the design problem (See Cross, 1984, for a review). Because this dissertation focuses on understanding information sharing in design, I selectively review the design literature on design activities to identify challenges in interdisciplinary collaboration. The specific aim is to illustrate that information sharing is crucial to design collaboration.

Different worldviews in design collaboration

Design team members bring in disciplinary differences. These disciplinary differences include, for example, different worldviews (thought), different ways of working (processes), different languages (communication), and different systems (tools). Therefore, disciplinary groups have their own way of thinking and working, and that they have discipline-specific worldviews of a design situation, which could make collaboration more difficult (Bucciarelli, 1994; Dougherty, 1992; Arias et al., 2000; Fischer, 2001; Sonenwald, 1995). Different worldviews often result in tensions and negotiation during the design process, and can create a barrier to interdisciplinary collaboration. Bucciarelli (1994) conducted a sustained ethnographic study of engineering design processes in two engineering firms and discovered that different design engineers worked within different *object worlds*. That is, due to a difference in disciplinary background and professional experiences, each engineer creates his/her own view of the object (i.e. a particular artifact, technical problem, or process) and sees it differently.

The concept of object worlds is similar to the concept of *thought worlds* (Dougherty, 1992). The term "thought world" refers to interpretive schemes, which was used to explain how different functional departments focused on different aspects of situations, and therefore, had different view of the whole. According to Dougherty (1992), different thought worlds are considered an interpretive barrier to successful product innovation. Similarly, based on a study of a long-term collaborative design project, Arias et al. (2000) used the term *culture* to explain that individuals involved in collaborative design use different norms, symbols, and representations because they come from different cultures, which often results in communication breakdowns. Fischer (2001) described different worldviews in terms of communities of practice, where stakeholders of design projects came from different communities of practice, and thus used different languages and different knowledge systems.

Interdisciplinary collaboration process

Design team members have to discuss and negotiate when collaborating across boundaries of these disciplinary differences in order to solve design problems. Individuals need to communicate their worldviews (i.e. discipline specific knowledge, interpretation, and value) to others as well as to explore and incorporate other worldviews. Bucciarelli calls this process a social process of negotiation, through which collaborating engineers with different object worlds meet, agree, and harmonize their ideas and concerns. Sonnenwald (1995) introduced the concept of *contested collaboration* to describe this process while Boland and Tenkasi (1995) called it the process of *perspective making and perspective taking*. Sonnenwald (1995) studied the collaboration between users, designers, and developers of information system design projects and found that participants from these different groups often challenge each other's ideas and contributions as a way to gain an understanding of one another's worldviews. Boland and Tenkasi (1995) explained that in order for cross-functional teams to collaborate, the team members must go through the process of perspective making and perspective taking. That is, team members need to first communicate their unique knowledge, and then be able to incorporate the knowledge from others. Along the same line, Detienne (2006) used *the confrontation and combination of perspectives* to describe the negotiation process among disciplines, in which designers from various disciplines clarified their design rationales and agreed on design solutions.

Different design artifacts created and used in design collaboration

Artifacts play an important role in design collaboration, especially in the interdisciplinary design context. Design team members use artifacts to support negotiation and discussion among disciplines. Because each disciplinary group uses its discipline-specific tools and creates different representations, design team members typically interact with a variety of artifacts during a design process (Bucciarelli, 1994; Perry & Sanderson, 1998; Schmidt & Wagner, 2004; Vidal et al., 2004). Perry and Sanderson (1998) observed that design engineers used a wide range of artifacts during the design process. Such artifacts include *design artifacts* (e.g., sketches, plans, models, and prototypes) and *procedural artifacts* (e.g., forms, office memos, letters, Gantt charts). Design artifacts are representations of design ideas while procedural artifacts represent design process. They reported that designers often used these artifacts in a collaborative manner. For example, designers collaboratively modified a drawing during a design review. Their findings suggest that design and engineering work is constructed through the interactions of multiple actors and that artifacts play an important role in the organization of design work.

Henderson (1991) conducted a participant observation to study the role of visual representations in the daily practices of design engineers. Her study focused on the introduction of a Computer-aided design (CAD) system. She discovered that sketches and drawings enabled design engineers to communicate and incorporate knowledge between different groups in the design. In addition, she reported that the use of CAD system reduced the flexibility necessary for drawings to function as an interactive communication tool that facilitates differential readings by different groups of users. Subrahmanian et al. (2003) investigated the use of prototypes in engineering design and found a similar issue, namely that the change in technologies disrupts the role of artifacts used to support design collaboration.

Schmidt and Wagner (2004) conducted an in-depth ethnographic study of coordinative practices and artifacts in architectural design and planning over the course of five years, and found the architects' workspaces filled with various artifacts such as plans, sketches, physical and virtual models, newspaper clippings, notes, photographs, faxes, books, and samples. They pointed out that some artifacts are produced for design purposes, while others are for coordination purposes. The study particularly investigated how architects use these artifacts, to coordinate their work. They concluded that architects had specialized practices and artifacts, which were designed to address specific coordinative issues and concerns.

Information sharing is crucial, yet challenging in interdisciplinary design

Information sharing has been recognized as a crucial part of interdisciplinary collaboration. Interdisciplinary design has made information sharing difficult because members from different disciplines bring to the team different design perspectives, different design languages, and different design methods, different information needs, different design representations, and different tools to create design artifacts. All these differences, thus, lead to the difficulty in coordinating and collaborating across disciplines. Curtis et al. (1988) conducted a field study of 17 large software system design projects. Their findings revealed that three most salient problems in software system design were the thin spread of application domain knowledge, fluctuating and conflicting requirements, and communication and coordination breakdowns. They found that design teams spent a large amount of time during early design phases defining terms, coordinating representational conventions, and creating channels for information flow.

The challenges in information sharing faced by interdisciplinary design teams can be summarized as follows:

1) Interdisciplinary design teams, bringing together many unique perspectives, engage in a negotiation process, in which team members communicate their own perspectives as well as integrate other's perspectives in order to solve design problems.
2) Differences in disciplinary background can hinder a clear communication and effective information sharing. Team members from different disciplines may view and solve the same problem from different perspectives, with their own unique method and language, which may create barriers to information sharing.
3) Interdisciplinary design team members create and exchange a wide range of design artifacts such as sketches, diagrams, specifications, and CAD models to support coordination and communication among different disciplinary groups. These design

artifacts are sometimes referred to as design representations (Henderson, 1991; Hendry, 2004; Reeves & Shipman, 1992).

In conclusion, information sharing is an important activity in design collaboration; however, assuring successful information sharing is challenging and questions still remain. For example, how does information get shared across disciplines during the design process? How do design teams use information technologies to support information sharing? To help answer these questions, I now review literature relevant to information sharing in design.

2.2 Information Sharing in Design

A number of research studies suggest that information sharing is essential to collaboration, especially in a highly interdisciplinary area; the success of collaboration depends on information sharing among the members (Palmer, 2001; Sonnenwald & Lievrouw, 1997). Interdisciplinary design team members typically experience an increasing amount of information from their own disciplines as well as new information from other disciplines. It can, therefore, be difficult for interdisciplinary design team members to share what they know with each other, especially with people from a different discipline. To improve design collaboration, information sharing across disciplines needs to be supported. However, before I get into information sharing in design, it is important to understand what "information sharing" actually is.

2.2.1 Understanding "Information Sharing"

Let me start by discussing the relationship between "information" and "knowledge." Information and knowledge are closely related. Often, these two terms are used interchangeably. Therefore, to understand information sharing, the meaning of "information" and its relationship to "knowledge" need to be clarified. The term "information" has been used to refer to a wide variety of entities, such as books, web pages, electronic files, photographs, and ideas. Various definitions of the term "information" have been proposed and applied in the field of information science (Meadow & Yuan, 1997). I am particularly interested in Buckland's definition of information: information-as-process, information-as-knowledge, and information-as-thing (Buckland, 1999). However, I do not agree with his notion of information-as-process because the act of informing or telling is not what is usually referred to as "information."

For the purpose of this dissertation, **I define information broadly as any informative entities, both tangible (e.g., books, documents, and multimedia files) and intangible (e.g., ideas, feedback, opinion, and comments).** This definition is adequate to account for a wide range of what people share during information sharing processes.

Next, I explain how I define the relationship between "information" and "knowledge." One view that has been widely used to distinguish "information" and "knowledge" is a hierarchy of data, information, and knowledge, which is often described as a series, starting with data converted to information, information converted to knowledge, and knowledge converted to wisdom (Ackoff, 1989; Boisot & Canals, 2004; Davenport & Prusak, 2000). However, this view is challenged by Tuomi's alternative view of the reversed hierarchy, which suggests that "data emerge only after knowledge and information are available" (Tuomi, 2000, p. 107). According to these views, knowledge seems to be more valuable and contextualized, and attached to persons than information. But, the distinction and relationship between "information" and "knowledge" is still unclear, in particular when considering the notion of explicit and tacit dimensions of knowledge.

Given all of this, therefore, I define the relationship between information and knowledge as follows: "information is a representation of codified explicit dimension of knowledge." The reason is that this definition reflects my interpretation of what knowledge actually being shared through information sharing processes. According to Polanyi (1983), knowledge consists of two inseparable parts: explicit dimension and tacit dimension. The explicit dimension of knowledge can be expressed, codified, and captured, while the tacit dimension of knowledge can only be known or understood through one's own practice or experience. Furthermore, I agree with Cook and Brown (1999) that one dimension of knowledge cannot be converted into the other. That is, tacit knowledge cannot be turned into explicit knowledge and vice versa. If the knowledge is expressed or verbalized, that part of knowledge is certainly not tacit.

Information sharing has been a subject of study by researchers from various communities, though they focused on different contexts, different aspects and different types of sharing. Understanding how various disciplines have investigated information sharing can shed some light on characteristics of information sharing. Thus, I review the literature related to information sharing in work contexts in the following areas: (1) human information behavior, (2) knowledge sharing, and (3) computer supported cooperative work (CSCW).

Human information behavior: Information sharing as human information behavior

Since researchers have given different definitions to the term information behavior (Pettigrew et al., 2001), a definition of information behavior is needed before I go any further. Pettigrew, Fidel, and Bruce define information behavior as *"the study of how people need, seek, give, and use information in different contexts, including the workplace and everyday living"* (Pettigrew et al., 2001, p. 44), while Wilson defines information behavior as *"the totality of human behavior* *in relation to sources and channels of information, including both active and passive information seeking, and information use"* (Wilson, 2000, p. 49). For the purpose of this dissertation, I define information behavior as human interaction with information – need, create, seek, manage, use, and disseminate information – in various contexts. The term "interaction" is used to convey a broader meaning than just seeking, need, and use. Thus, information behavior studies in design include any studies that seek to understand how designers interact with information during design processes.

The focus of Information Behavior research in Information Science has been largely on information seeking, needs, and use, while little attention is given to information sharing. Information sharing has often been investigated with regard to information seeking. That is, information sharing occurs as part of or a consequence of information seeking. Occasionally, people share information as a strategy to obtain needed information. For example, Poltrock et al. (2003) found that software and hardware designers disseminated ideas or documents in order to get feedback from others.

Research studies have revealed that people shared information they found, intentionally or unintentionally, with others. For example, O'Day and Jeffries (1993) studied the use of search results by library clients. All participants in their study shared the search results they found with their colleagues. They discovered four basic models of sharing – updating (passing the information on to specific others), consulting (sharing information in response to a request from others), broadcasting (circulating information to a bigger group of people), and archiving (putting potentially useful information into a shared repository). Erdelez and Rioux (2000) investigated how people shared information they accidentally found on the Web and anticipated that it might be of interest to others. They found that some people shared encountered information by sending an email message containing a cut-and-paste URL of interest and some personal and contextual information. Marshall and Bly (2004) studied sharing practices for encountered paper and electronic publications in the course of every life.

Knowledge sharing: Information sharing as part of knowledge sharing

Knowledge sharing has been recognized as a source of innovation for organizations (Carlile, 2002; Connelly & Kelloway, 2003). Research in this area is concerned with how organizations can promote knowledge sharing among employees (interpersonal), among teams (intra-organization), and across organizations (inter-organization). Notable findings from this research are that organizational factors, technological factors, and personal factors influence knowledge sharing at the interpersonal level while intra- and inter-organization knowledge sharing are largely influenced by cultural factors and organizational factors (Connelly & Kelloway, 2003; Mesmer-Magnus & DeChurch, 2009; Majchrzak et al., 2000; Vatrapu & Suthers, 2010; Yang & Maxwell, 2011). Through a survey study, Connelley and Kelloway (2003) found that employees' perceptions of social interaction culture and management's support for knowledge could predict their perceptions of knowledge sharing culture of the organization. Some researchers particularly explored knowledge sharing across disciplines or specializations using the concept of boundary objects. Carlile (2004) proposed an integrative framework for examining knowledge sharing across boundaries, and Gasson (2005) examined knowledge sharing across boundaries through computer-mediated collaboration.

Knowledge sharing has also been studied in relation to the concept of shared mental models. The concept was first introduced by Cannon-Bowers and Salas to describe "knowledge structures held by members of a team that enable them to form accurate explanations and expectations for the task, and, in turn, to coordinate their actions and adapt their behavior to demands of the task and other team members" (Cannon-Bowers et al., 1993, p. 228). The definition infers some kinds of shared knowledge between team members. Researchers have used several different terms to refer to similar concepts – team mental model, shared cognition, and transactive memory (Cannon-Bowers & Salas, 2001; Langan-Fox et al., 2004; Mohammed et al., 2000; Moreland, 1999). Researchers in the area of team performance have, for a long time, used the concept of shared mental models to explain and predict team performances. However, this concept is not directly relevant to understanding information sharing processes, which I want to explore. The shared mental model focuses on known information while this dissertation focuses on sharing new information, not previously shared, as in a shared mental model. Therefore, I have decided not to use the concept of shared mental models in this dissertation.

Computer supported cooperative work: Information sharing systems

Unlike the subfields of human information behavior research and knowledge sharing, which have taken a generally neutral stance on technology, research in CSCW has been focused on collaborative technologies or at least the coupling of people and technologies. Much cooperative work is conducted by large heterogeneous groups of people. In such work settings shared information spaces often become essential means for the collaborative activities (Carstensen, 2000). Thus, a number of CSCW researchers have been interested in the notion of shared information spaces for collaborating teams (Bannon & Bødker, 1997; Berlin et al., 1993; Bossen, 2002; Davis et al., 2001). Berlin et al. (1993) developed TeamInfo, a shared repository using email-based input. Team members could browse, search, or ask to be notified of new items. The challenge faced in developing TeamInfo was to enable low-effort sharing while remaining individuals' finding effectiveness. The result of a pilot use of TeamInfo revealed a serious problem on shared information structure. That is, different filing styles caused the team members to classify the same items differently, which made it difficult to retrieve shared information (Berlin et al., 1993).

Taking the notion more conceptually, Bannon and Bødker (1997) introduced the concept of common information space. According to Bannon and Bødker (1997), common information spaces are the information spaces that are implicitly or explicitly co-constructed and shared by cooperating actors with the aim of interpreting and articulating their work. It is important to note that these spaces do not just consist of objects, events, or entities in the shared space, but they also engage the joint interpretation of these objects, events, or entities by the actors involved (Bannon & Bødker, 1997). Bossen (2002) later refined the concept by proposing parameters of common information spaces based on the ethnographic fieldwork in a hospital ward.

Some CSCW researchers have focused on sharing specific types of information (Davis et al., 1999; deSouza et al., 2003; Goecks & Cosley, 2002). Goecks and and Cosley (2002) developed NuggetMine, an intelligent groupware application, to support asynchronous sharing of information nuggets among the group. Information nuggets were defined as *"small amounts of self-contained information, such as the URL of an interesting news article, a book title, or the time and location of a local art event"* (Goecks & Cosley, 2002, p. 87). Davis et al. (1999) developed an ink-based, collaborative note-taking application called NotePals. NotePals allowed team members to easily share hand-written notes. deSouza (2003) examined how software developers shared their work, pieces of software code, with others. To share the work here means to make it publicly accessible for other team members. The study revealed that developers employed a complex set of practices, together with a number of tools, to handle the transition between private and public work (deSouza et al., 2003).

Defining information sharing

The review of studies above emphasizes the fact that information sharing is everywhere and that information sharing activities can differ in objects and methods of sharing. People share a variety of information, ranging from intangible information (e.g., ideas and knowledge) to tangible information (e.g., hand-written notes, search results, and information nuggets). They also employ different methods of sharing, for instance, face-to-face interaction, email, and using a shared repository.

Based on my review above, I conclude that information sharing always involves at least two actors: a sharer and a receiver. Furthermore, information sharing is not completed until a receiver gets the shared information. The sharing may occur between one sharer and one receiver, one sharer and multiple receivers, multiple sharers and one receiver, or multiple sharers and multiple receivers. During the information sharing process, the actors may engage in multiple aspects of information sharing, such as sharing the access to information, sharing an information artifact, sharing the content, and sharing meaning.

2.2.2 Information sharing studies in interdisciplinary design

Many researchers have studied information sharing in design, though focusing on different aspects of sharing. That is, some researchers are more interested in sharing information artifacts, while others are more interested in understanding the sharing of meaning behind shared artifacts. Thus, I grouped the studies on information sharing in design based on the focus of studies to demonstrate the fact that many studies on information sharing in design have focused on particular aspects of information sharing activities.

Sharing information artifacts

The studies focusing primarily on the sharing of information artifacts include studies on shared information spaces and how people share design artifacts. Shared information spaces have been recognized as important to collaborative design work (Bannon & Bødker, 1997). Many design projects have a shared information space (also called a common information space or a shared repository) where project members share information. A shared information space can be an electronic space (e.g., a shared directory or folder on a server) or a physical space (e.g., a filing cabinet). It seems though that electronic spaces are becoming more popular than physical spaces as a lot of information is created and captured electronically (Kumar et al., 1994; Trigg et al., 1999).

A number of researchers have studied how shared information spaces are (or could be) created, used, and maintained (e.g., Bossen, 2002; Carstensen, 2000; Lahti et al., 2004; Reeves & Shipman, 1992; Trigg et al., 1999). For example, as part of their study on migrating paper documents to an online repository, Trigg et al. (1999) examined document filing practices of design engineering teams. They found that the problem occurred when individual filing approaches made it difficult for others, who have different viewpoints, to retrieve documents, or when groups had incompatible views of the document body. Similarly, Carstensen (2000) examined how the common archives of a large engineering design project were established, structured, maintained and accessed by a team of engineers who had different backgrounds and expertise. The findings of this study were that (1) the structures in the centralized archives evolved overtime, (2) the classification schemes applied were very simple (most entries were organized into only two levels of categories), and (3) the content of a group common information space was accessed by many different actors having different backgrounds, expertise and needs.

One issue that arises with the use of shared information spaces in interdisciplinary design projects is that information organization differs among different disciplinary groups and that individual information access strategies break down with group information stored in the shared information space (Bannon & Bødker, 1997; Berlin et al., 1993). Davis et al. (2001) discovered this issue when conducting a study of information flow and use of product design teams to develop shared information spaces. They found that different functional departments such as marketing, manufacturing, and engineering support services exchanged information through phone calls and internal mail system. There was no single central accessed to the project information. In addition, different groups organized design documents differently, which made information sharing across departments more difficult. Based on their findings, Davis et al. (2001) developed a shared information space that could adapt to the changing needs of the users. That is, information could be stored and viewed in different formats, and organized and accessed using multiple classifications by multiple perspectives. Subrahmanian et al. (2000) also found the same issue in their study of the use of classification scheme in engineering design. They found that disintegration of classifications resulted in breakdowns in consensus and conflict resolution. The study illustrated that a simple classification system that evolved through incremental modifications was not sufficient. A more rigorous organization-wide classification was needed to map classification schemes used by different groups – design, marketing and manufacturing.

Another issue about the use of shared information spaces is concerned with moving information between a private space and a shared (or public) space. People have different strategies to release their private information, and also expect the same from others. That is, people should not overload others with public information that is not relevant to their current work. Geisler & Rogers (2000) indicate that one of the basic characteristics of design collaboration is the interweaving of public and private spaces. Therefore, with a shared information space, design team members want to decide when and how to share their information. With paper documents and a physical shared space, people have a complete control of sharing information with others; they can keep it until they feel that the information is ready to be shared. In contrast, with an electronic shared space, people may not have much control because the information may always be available to others. For example, when using the configuration management tools, software engineers selectively choose when and how to disclose their private work to others (deSouza et al., 2003). The use of the configuration management tools created a clear distinction between the private aspects of work developed by a software engineer and the public aspects when sharing his work with others. deSouza and his colleagues found that software engineers developed formal and informal work practices to gain more control of sharing information. Thus, the study concludes that it is necessary to support transitions between private and public.

Focusing on how design teams share information artifacts and messages, Poltrock et al. (2003) studied collaborative information retrieval and information sharing of software and hardware design teams. They found that despite differences in design products, disciplinary backgrounds, and tools, the two design teams employed similar information seeking strategies during their collaborative information retrieval. Moreover, information sharing was found as part of collaborative information retrieval.

Sharing the meaning

Instead of focusing on the sharing of the tangible aspect of information, many researchers have focused on the original meaning that people want to communicate when sharing information. Therefore, information sharing has also been addressed as part of communication. Communication is problematic in interdisciplinary design because each disciplinary group brings in different viewpoints, knowledge, and expertise (Yamaoka et al., 1998), which in turn create different interpretations of shared information. Grudin and Poltrock (1989) conducted a survey on coordination and communication across disciplines in user interface design and found that communication among the different professionals and managers were problematic. Results suggested the need to facilitate members of the two groups to work more closely and to share expertise more easily.

Design teams employ various techniques in supporting communication including using technologies, artifacts, and mediators. Sonnenwald (1995) observed different communication roles assumed by design team members. These roles – agent, external star, intergroup star, gatekeeper, and boundary translator – affect information sharing at different levels. Gatekeeper and boundary translator are of particular interest. Gatekeepers filter and/or block information originating outside the group to/from group members while boundary translators explain group information to others who are not members of the group (Sonnenwald, 1995).

Various artifacts (e.g., drawings, diagrams, sketches, office memos, letters, and other design documents) are used as a form of communication during a design process (Curtis et al., 1988; Perry & Sanderson, 1998; Sonnenwald & Iivonen, 1999). One concept that often used to describe the role of design artifacts in supporting coordination across groups of individuals with different interests, perspectives, and/or ways of thinking is boundary objects. The notion of boundary objects was developed as a structure for coordinating distributed work. The concept was developed by Star to refer to "objects that are both plastic enough to adapt to local needs and constraints of the several parties employing them, yet robust enough to maintain a common identity across sites." (Star, 1989, p. 46) Artifacts, documents and perhaps even vocabulary or common language that help people from different communities build a shared understanding can be considered boundary objects. As boundary objects, design artifacts, in particular drawings and sketches, are used to support design communication (Henderson, 1991; Perry & Sanderson, 1998) and to coordinate design activities (Hertzum, 2004; Perry & Sanderson, 1998; Schmidt & Wagner, 2004).

Hendry (2004) identified five communication functions of design representations. Those functions are: conscripting, coordinating, framing, persuading, and recording. Because a lot of artifacts, both in electronic and paper forms, were exchanged, transferred, and disseminated among design team members as a means of communication, artifact use can then reveal communication patterns of design teams. The studies that investigated document use as a communication channel typically focused on patterns of communication and information flows during the design process. For example, Sonnenwald and Iivonen (1999) examined documents distributed among design groups to identify the content, the frequency and direction of information dissemination and communication during design process. Similarly, Davis et al. (2001) investigated the exchange of artifacts as part of the information flow analysis of product design teams.

In some cases, designers exchange or disseminate artifacts to gather information (e.g. feedback and comments) from others (Hendry, 2004; Poltrock et al., 2003). The exchange of one artifact may sometimes lead to a creation of another artifact. In other cases, designers may distribute artifacts just to broadcast information (Perry & Sanderson, 1998). When a number of artifacts, especially documents, circulate during the design process increases, version control becomes an issue, more specifically how to ensure that only current versions are in used. Perry and Sanderson (1998) found that document version control was a major problem in the engineering design project they observed. The design team spent a lot of time and effort reviewing the circulating documents to ensure that only appropriate documents were circulated.

Although many studies reported that artifacts were often used to support communication (Boujut & Blanco, 2003; Hendry, 2004; Römer et al., 2001), relying on shared artifacts as a sole communication medium might not be effective. Curtis et al. (1988) found that design documents were used in all levels of communication – individual level, team level, project level, organizational level, and inter-organizational level. However, shared documents did not provide sufficient communication because of their tardiness, incompleteness, and lack of interaction, which led to communication and coordination breakdowns.

Sharing knowledge and expertise

Many researchers are interested in how knowledge and expertise is shared when design team members engage in information sharing activities. Previous research shows that individual knowledge not only needs to be shared among design team members, but it also needs to be integrated in order to solve design problems. That is, it is not sufficient to know what others know, but also to know how to apply what is known. Walz et al. (1993), focusing on requirements specifications and design, investigated how software design team members acquire, share, and integrate project-relevant knowledge. The data were collected from team meetings. The study found that: (1) the design team acquired, shared, and integrated the knowledge necessary for the design task in order to get up to speed; (2) the design team integrated the necessary knowledge into a shared understanding of the application and the design in order to create the team memory; and (3) team members participated unequally in different stages. Similarly, Sole and Applegate (2000) investigated how dispersed, cross-functional development teams exchanged and combined task-relevant knowledge using collaborative technologies. Faraj and Sproull (2000) focused specifically on expertise sharing in software development teams. They define expertise as "the specialized skills and knowledge that an individual brings to the team's task" (Faraj & Sproull, 2000, p. 1555). They concluded that to effectively share and coordinate expertise, team members must: know where expertise is located, recognize the need of expertise, and bring in expertise to solve the problem in timely manner.

Some researchers have examined specifically how knowledge is integrated. Recent organizational research has pointed out that knowledge integration is crucial in collaborative design, in particular for new product design and development (Bechky, 2003; Carlile, 2004). The knowledge integration process is also called "knowledge transformation," through which individuals modify some of their existing knowledge as they come to understand how to utilize knowledge from another disciplines in the context of their own work (Bechky, 2003; Carlile, 2002; Kellogg et al., 2006; Levina, 2001). Two ethnographic studies were conducted to investigate how knowledge is transformed across disciplines in a new product development process. Through a study of product development teams, Carlile described knowledge as localized, embedded, and invested in practice carried out by each discipline, which makes knowledge transformation difficult (Carlile, 2002). He found, however, that product development teams used the collection of artifacts to represent, learn about, and transform knowledge across disciplinary boundaries. Similarly, Bechky (2003) explained that knowledge sharing across occupational groups was difficult due to the differences in their language, their practices, and their conceptualization of the product. She focused particularly on the process of knowledge transformation that occurred between engineers, technicians, and assemblers involved in the production. Misunderstandings between these different groups were reconciled through the use of "tangible definitions," a physical demonstration that combines verbal explanation with tangible objects.

Multiple aspects of sharing information

While many of studies reviewed above focus primarily on one aspect of information sharing, one study has recognized information sharing as a multi-process activity. Citera et al. (1995) examined collaboration in interdisciplinary design teams, focusing on information sharing. They described internal processes and external obstacles of information sharing. The internal processes of information sharing were: reciprocal exchange of information, establishing trust and credibility, establishing shared understanding, and compromise and negotiation. The external obstacles to effective information sharing were: institutional disincentives, historical and ideological barriers, power disparities, differing perceptions of risk, technical complexity, and political and institutional culture.

Lessons learned from previous studies on information sharing in design

Although previous studies help us better understand information sharing phenomenon, most studies on information sharing in design have focused on particular aspects of information sharing activities (e.g., sharing access, sharing artifacts, sharing meaning, or sharing knowledge). For example, the communication view often used to study sharing of meaning focuses primarily on information sharing activities whose goal is to support a communication act. It does not account for other types of information sharing where the act of sharing may not lead to communication (e.g., allowing others to access one's information, disseminating information just for FYI). I argue that it is important to investigate information sharing from a holistic view, which looks at multiple aspects of information sharing. That is, to try to understand what exactly people do when they engage in "information sharing," why they do that, and the outcomes of the sharing. This study examines information sharing from such perspective.

Moreover, many tools have been developed and used to support information sharing in design. The use of WWW technology is becoming increasingly important for the sharing of information within design organizations. Tools that support a more collaborative approach to content creation and management that may be more suitable for design teams include instant messenger, blogs and wikis – server software that allows users to freely create and edit web page content and organization using any web browser and on the fly. Particularly, the use of wikis appears to be growing rapidly. Both academic and industrial communities are interested in how and why wikis became so popular, and how they will be used in the future. There are at least two international events devoted to just wikis – WikiSym, an annual symposium sponsored by ACM, and Wikimania, an international conference sponsored by Wikimedia organization. In the next section, I review literature relevant to the wiki technology.

2.3 Wiki

2.3.1 Introduction to wiki

Wikis are a multi-user tool that collect and cross-reference information (Leuf & Cunningham, 2001). The term "wiki" is commonly referred to as both wiki websites and the software used to create them (e.g., Klobas, 2006; Wagner, 2004). Ward Cunningham, a wiki pioneer, describes a wiki as, "the simplest online database that could possibly work" (Leuf & Cunningham, 2001). With simple text syntax, new wiki pages and crosslink between pages can be created on the fly. Unlike ordinary websites, wikis allow all readers to easily create and edit the content as well as the structure of Web pages online using any Web browser. Generally, changes are accepted without being reviewed. This "open editing" concept of wikis has diminished a line between Web page's authors and readers. That is, any reader, particularly non-technical people, can become an author of a Wiki site.

WikiWikiWeb, the first-ever wiki, was developed on 25 March 1995 to promote idea exchange in a software developer community. It was developed to supplement the Portland Pattern Repository to publish pattern languages and pattern related information. "Wikiwiki" is a Hawaiian word that means quick, speedy, or fast. Therefore, "WikiWikiWeb" name emphasizes its key characteristic, with which its content can be made available and changed in a quick, easy manner. Wikis' original design principles (http://c2.com/cgi/wiki?WikiDesign Principles) are: open, incremental, organic, mundane, universal, overt, unified, precise, tolerant, observable, and convergent.

Since the first wiki, over a hundred wiki engines have been deployed (See http://www.wikimatrix.org/ for a list of wiki engines), so called "wiki clones," developed mostly under open source licenses, which makes them available for free use and editing. All follow the same fundamental design principles but are different in the underlying technology and the assumptions about their use. For example: TWiki (http://twiki.org/) is an enterprise collaboration platform and knowledge management system, which aims at corporate use. It is more structured than WikiWikiWeb, the original wiki, as it offers templates and forms and allows users to incorporate their workflow into the wiki site; MediaWiki (http://mediawiki.org/), a free wiki software package, is originally written for the Wikipedia project; CoWeb is a collaborative Web site used to facilitate classroom collaboration (Guzdial et al., 2001).

Key features of wikis

Social software refers to computing tools that facilitate social interaction, communication, collaboration, and information exchange among groups of people (Klobas, 2006). Social software includes, for example, instant messaging programs, bulletin boards, weblogs, listserv (or mailing lists), music and photo sharing tools, and, of course, wikis. Thus, wikis are often reviewed, along with or in comparison to, other social software (e.g., Avram, 2006; Fichter, 2005; Mattison, 2003; Wagner, 2004). So, how do wikis differ from other social software?

Open editing is a key concept that makes wiki technology unique. That is, a reader is also a potential author of the content. In fact, wikis seek to involve their visitors in a process of creation and collaboration that constantly changes the wiki site content (Leuf & Cunningham, 2001). Moreover, wikis are unusual among other group communication tools in that they allow the organization to be edited in addition to the content itself.

Key features of wikis are summarized below:

- Easy access: Wiki sites consist of pages that are viewable in a web browser.
- Easy editing: Wiki pages can be edited by anyone using a web browser on any computer connected to the Internet. This is a feature that encourages the visitor to be involved in an ongoing process of creation and collaboration that constantly changes the wiki content.
- Simple wiki syntax: Wikis allow users to do typical text formatting without the need to learn HTML. New users only need to learn a few formatting tags.
- Links: Page links are designed to be very simple to create. There are no broken links to internal content because all pages that do not yet exist will have an empty page that users can immediately edit.
- Version history: Wikis provide the ability to compare previous versions of a wiki page, which provides details on who edited what part and when. In addition, previous page revisions can be retrieved, as a mechanism to protect vandalism, spam or errors.
- Discussion space: Wikis often have a space for users to discuss any issues regarding changes made to each wiki page.
- Search: Most wikis offer at least a title search, while some have full-text search.

Use of wikis

Wikis have been used for many different purposes, ranging from individual use, small-scale collaboration, to large-scale collaboration. For personal use, Leuf and Cunningham (2001) highlighted that wikis could be great personal information managers because every piece of information (e.g., notes, files, comments, contacts, and resources) will eventually be interconnected with an unlimited number of cross-links created through wikis. However, wikis are often thought of as a tool for multiple authors rather than a single author. Wikis can be used as an online collaborative space for groups or teams working on a project. They sometimes replace groupware (Klobas, 2006). Common use of wikis for group collaboration include, but are not limited to, creating, sharing, and storing project documents, tracking revisions, and meeting planning. These group-collaborative wikis usually have restricted access to group members only.

For large-scale collaboration, wikis have been used, for example, to co-author websites by communities of practice or special interest groups, to jointly develop shared directories or information repositories, and to plan conferences and events. The most well-known, successful example of publicly collaborative wiki is probably Wikipedia, the largest online encyclopedia. Started in 2001, Wikipedia now has more than 18.6 million articles in 250 languages, contributed by over 5 million users, and viewed by over 381 million unique visitors worldwide (http://stats.wikimedia.org/reportcard/). Although the open editing concept raises some concerns about vandalism, misuse, and reliability of the information collaboratively created by multiple users, many public wikis have proved not to be the case.

The success of public wikis has made businesses start to look at wikis as a means to support online collaboration in the workplace. There has been an increase in corporate and intranet use of wikis over the past few years. Many organizations use wikis as a project collaborative tool, as a knowledge base, and as an internal information gathering and disseminating tool (Klobas, 2006). Wikis are also widely used in educational context to facilitate collaborative learning (e.g., Kussmaul et al., 2006; Lamb, 2004; Malani & Dwyer, 2005). In fact, wikis have been claimed to be a powerful web tool that teachers and instructors can use to promote student learning and collaboration (Guzdial et al., 2001; Kane & Fichman, 2009; Larusson & Alterman, 2009; Parker & Chao, 2007).

Notably, the use of wiki sites varies depending on the goals of users – the group, the community or organization – that utilize it. Wiki sites can be publicly available for anyone to read and modify on the Internet, or they can be private, with restricted access, to support collaboration among a small group.

2.3.2 Wiki: The development and evolution of the technology

Since its first appearance, wikis have received a lot of attention from the software development community on developing wiki clones, augmenting the wiki technology, and incorporating wiki software with other tools. Wiki clones were often developed to accommodate specific needs in different contexts. For example: SmallWiki, an object-oriented wiki was developed as a collaborative content management system because, it is claimed, the original wiki implementation is not flexible and powerful enough to support collaborative content management (Ducasse et al., 2005); CoWeb, a collaborative Website, was developed and placed into classroom use at the Georgia Institute of Technology to support teaching and learning activities (Guzdial et al., 2001); to solve a documentation problem in software development projects, Aguiar and David (2005) developed XSDoc Wiki to integrate documentation produced by software developers. Many found a way to incorporate wiki with other tools. Eto et al. (2005) developed qqikWeb system by combining wiki and a mailing list to support group communication. Adding wiki collaboration to single display groupware, Baraldi et al. (2006)

developed wikiTable system, which integrated two different collaboration contexts, face-to-face sessions and web-based knowledge space, to improve knowledge building experience.

Wikis are open source software so a lot of extensions to wikis have been developed. Interestingly, one type of wiki augmentation is to put more structure and control in wikis, which seems to contradict with the original design of wikis. While flexibility and openness is a key characteristic of wikis, many have argued that there is a need for more structure in wikis and have proposed technical solutions to create structured wikis (Burrow, 2004; Di Iorio & Zacchiroli, 2006; Haake et al., 2005). Since most wikis do not provide a nice printable version of wiki content, WikiPublisher was developed as a print-on-demand plug-in to allow wiki readers to convert wiki content into a high-quality printed document (Rankin et al., 2009). Some researchers have integrated wiki with other technologies such as file-sharing system, email and twitter to enhance its utility (Hanrahan et al., 2011; Voida & Greenberg, 2009; Zhao et al., 2011).

The work discussed in this section is only a small portion of all wiki development work. The wiki literature was selected to reflect a wide range of wiki's application and its development. It is interesting, however, to note that most of these development efforts were not informed by research into the use of wikis. Next, I discuss wiki literature that aims at investigating the use of wikis and/or user experience.

2.3.3 Wiki research

When considering the context of use, the majority of research studies fall into online communities (such as Wikipedia), educational settings, and corporate settings. First, I give an overview of research studies in each setting. Then I review research studies related to corporate project wikis with a focus on the use of wikis to support software design and development projects.

Wikipedia and online community wikis, are the most popular context of study with 485 journal and conference articles from ACM digital library with “Wikipedia” in its title (as of September 30, 2011). With Wikipedia being the largest free online encyclopedia *anyone can edit,* with anyone being anyone with Internet access, the majority of Wikipedia research has focused on contributors' coordination, the quality of content, vandalism, and trust and credibility. Wikipedia contributors tend to communicate via dedicated discussion pages before they edit the Wikipedia content, which reveals coordination and conflict patterns among contributors (Kittur et al., 2008; Viégas et al., 2004; Viégas et al., 2007). The same coordination and conflict patterns via discussion pages are also found in other online community wikis (Kittur & Kraut, 2010). Because of the openness of Wikipedia editing process, research studies have found that vandalism, trust, and credibility has become issues for the Wikipedia community, and have addressed these issues by developing mechanism to detect vandalism as well as to increase credibility of Wikipedia content (Geiger & Ribes, 2010; Javanmardi et al., 2011; Kramer et al.,2008; Mola-Velasco, 2011; Oxley et al., 2010; Pirolli et al., 2009; Potthast et al., 2008; Priedhorsky et al., 2007; Wang & McKeown, 2010). Based on their contribution, Wikipedia contributors have been categorized based on the amount of contribution (e.g., active users, infrequent editors, and readers) or the type of contribution (e.g., administrators, editors, and cleaners) (Antin, 2011; Liu & Ram, 2009; Panciera et al., 2009). Panciera and her colleagues found that most Wikipedia contributors made consistent contributions over time (Panciera et al., 2009). That is, the amount of individual wikipedian's contributions did not increase or decrease. Although ownership of content is discouraged within Wikipedia, several studies have found strong evidence of ownership behaviors in practice (Halfaker et al., 2009; Thom-Santelli et al., 2009). Interestingly, there is a gender gap in Wikipedia contributors. Two research studies found that the majority of Wikipedia contributors were men (more than 80%) and that men tended to make more revisions than women (Antin et al., 2011; Lam et al., 2011).

Educational wikis have long gained attention among educational researchers, who have been interested in exploring the use of wikis to support learning and classroom collaboration. The first study of educational wiki was dated back to 2001. Guzdial et al. (2001) reported on the use of CoWeb, a wiki-based application, by teachers and students. They discovered 25 kinds of activities invented from using CoWeb, which could grouped in three broad categories – distributing information activities; collaborative artifact creation activities; discussion and review activities. In general, wikis are perceived by students in various levels, including young children, to be easy to use by students (Desilets et al., 2005; Raitman et al., 2005). Wikis have been used in a classroom context to support student's collaboration, enhance student's learning, and encourage student's participation (Jaksch et al., 2008; Kepp & Schorr, 2009; Kussmaul et al., 2006; Reynolds & Caperton, 2009; Tselios et al., 2011). It is argued, however, that to maximize the utility of wikis in the classroom environment, wikis should be modified with certain features to address the special needs of the classroom (Larusson & Alterman, 2009; Wang et al., 2005).

Corporate wikis (also known as enterprise wikis) refer to the wikis developed and used in corporate settings. These wikis are often secured within the corporate firewall; not accessible to public. Following the success of Wikipedia, wikis have attracted many organizations in both public and private sectors. Wikis have been widely used for supporting various work activities

(Mader, 2008; Grudin & Poole, 2010; Majchrzak et al., 2006; Paquet, 2006). Popular applications of wikis in a corporate setting include a document repository, a collaboration tool, and a knowledge management tool (Arazy et al., 2009; Hester, 2010). Many researchers have focused on the use of corporate wikis as a knowledge management tool, and have found that wikis improve knowledge sharing and management in organizations (Hester, 2010; Grace, 2009; Sousa et al., 2010). The concept of visualization has been applied to enhance wikis to better support knowledge management. Hirsch and his colleague developed ThinkFree, an industrial Visual Wiki application, which provides an easier way for end users to navigate and understand enterprise architecture information in a large corporate wiki (Hirsch et al., 2010). VisualWikiCurator, an enterprise wiki plug-in, was developed to reduce the interaction costs of organizing and updating content on corporate wikis (Kong et al., 2011).

The majority of research on corporate wikis has focused on challenges, benefits, and success factors of corporate wikis, mostly through a survey study, and their findings suggest that the success of corporate wikis depend on user expectation, nature of information, information quality, existing information technology and practices, and corporate culture (Danis & Singer, 2008; Arazy & Croitoru, 2010; Grudin & Poole, 2010; Holtzblatt et al., 2010; Stocker & Tochtermann, 2009; White & Lutters, 2009; Yates, 2010). Previous research on corporate wikis also reveals that corporate wikis address different needs than public wikis. Unlike Wikipedia, the corporate wikis need a more complex access control to limit access to specific content (Danis & Singer, 2008). Moreover, vandalism has not been an issue in the corporate environment. A survey study revealed that corporate wiki users are motivated by making work easier, and helping the organization achieve its goals (Majchrzak et al., 2006). Díaz and Puente (2011) argued that corporate wikis need a more structure and perhaps pre-defined template for wiki content; therefore, they proposed wiki scaffolding as a way to align wiki activity with the organizational work practices.

Based on their empirical study, Poole & Grudin (2010) classified corporate wikis into three categories – enterprise pedias (corporate-wide information repository wikis), group or team wikis (project or special interests wikis), and single contributor wikis (personal wikis). Research on corporate wikis was often conducted based on the wiki usage of all corporate wiki instances, with a more focus on enterprise pedias. Egli and Sommerlad (2009) presented an experience report of wiki adoption in a law firm, where wikis were used to support knowledge management and collaboration. They observed that wiki instances often started as information-display or personal wikis, then developed into collaboration wikis. A study on the sustainability of corporate wikis using the time-series analysis of activity patterns was conducted on all wiki instances at IBM (Arazy & Croitoru, 2010). The study goals were to: develop a method for analyzing wiki activity logs; understand the patterns of corporate wiki revision activity; and examine the factors impacting corporate wikis' sustainability. The findings revealed that most corporate wikis became inactive after a relatively short period of time. However, it was unclear what those short-live wikis were enterprise pedias, group wikis, or personal wikis.

Research on project wikis

In general, wikis are used to support project documentation and serve as a project information repository. Wikis provide the ability to create and maintain documentation in a dynamic and collaborative manner (Radziwill & Shelton, 2004). Moreover, wikis also support the creation and record of both structured and unstructured knowledge (Chau & Maurer, 2005). Very few studies have investigated the use of project wikis. Danis and Singer (2008) reported the design and deployment and the 2-year use of Research Wiki for yearly planning work. The planning work required interdisciplinary collaboration between researchers, managers, strategists, funding specialists, and technical support staff. Through interviews and wiki log analysis, they found that the Research Wiki encouraged collaborative writing and received wide interest in the content shared in Research Wiki across disciplines. They also found that at least 60% of editors were not worried about the fact that their unfinished work could be seen by others, but editors were reluctant to modify others' content except in special circumstances. Similarly, Munson (2008) conducted a case study of the deployment of MediaWiki as a publishing tool for building organizational memory at Boeing and found that wiki group members did not feel comfortable making substantial edits to others' content but did occasionally use the wiki to coauthor content, and also categorize and link to others' content and fix typos.

White and his colleagues investigated the use of wikis within IT support groups and discovered two major impediments, which were concern over achieving a critical mass of content and anxiety over potential unintended or unexpected content change (White et al., 2009). Recently, a theoretical model for measuring corporate wikis' success from the end-user's perspective, particularly for project management task, documentation, and knowledge sharing, was proposed based on an existing information system success model (Bhatti et al., 2011).

While project wikis are prevalent in a corporate environment, especially, for design and development projects (Majchrzak et al., 2006; Munson, 2008; Poole & Grudin, 2010), the actual use of wikis and their usefulness in such collaborative situations is still little understood. Wikis are particularly beneficial to software design and development projects. Because of their flexibility, wikis have been used to support various software design and development activities such as project management, requirements gathering, design collaboration, and software documentation (Correia, 2010; Louridas, 2006; Wu et al., 2009). Most published research on the use of wiki in software design and development projects has been on supporting software documentation (e.g., Aguiar & David, 2005; Correia, 2010; Xiao et al., 2007), and software requirements engineering (e.g., Decker et al., 2007; Ferreira & Silva, 2009; Wu et al., 2009; Yang, 2009).

Wikis are found to support evolving documents and heterogeneous design artifacts produced by software design and development teams. To solve a documentation problem in software development projects, Aguiar and David (2005) developed XSDoc Wiki to integrate documentation produced by software developers. The XSDoc Wiki helps on weave different kinds of contents into a single heterogeneous document, while still preserving its semantic consistency. Similarly, WikiDev 2.0 and Weakly Wiki were developed to support software documentation. WikiDev 2.0 was developed to enhance collaboration within software development teams by increasing traceability of interdependencies among design artifacts produced by various individual tools (Bauer et al., 2009; Fokaefs et al., 2010). It integrates information from multiple tools and displays through its wiki-based front-end. The WikiDev 2.0 has been deployed at the University of Alberta for more than two years, and has been used by approximately 120 students taking project-based software-engineering courses; however, there was no a report or evaluation on its usage. Weaki Wiki, a wiki prototype especially designed to support incremental formalization of structured contents created in software development projects, was developed specifically to deal with evolving document structures, and increasing awareness of the content structure. WeakiWiki was applied in an academic software engineering lab. However, there was no report on its deployment and use (Correia et al., 2009; Correia, 2010). Xiao and colleagues developed Galaxy Wiki, an online collaborative software development environment, to enable software developers to write source code, compile, execute, and debug code in wiki pages. (Xiao et al., 2007).

Because of their flexibility, along with openness and collaborative authoring features, wikis have been used and enhanced to support software requirements engineering. Requirement engineering consists of requirements gathering and requirements documentation activities, which are complex due to the evolving nature of requirements themselves and also involved stakeholders. Decker and his colleagues (2007) experimented and investigated how wikis were used as a central technology to create software project requirements with a variety of stakeholders. They chose wikis because they provide a flexible platform that could support active stakeholders who differ in: perspectives on the developed system, backgrounds, objectives, abilities to express requirements, and involvements. The findings show that wikis are useful for stakeholder collaboration as well as for grouping and structuring requirements in software development projects.

Some researchers have enhanced wikis specifically to deal with the complexity of requirements engineering. WikiWinWin, was developed to support collaborative requirements negotiation between multiple stakeholders of a software development project (Yang et al., 2008). A retrospective case study of one student project in a software engineering class, conducted to evaluate the WikiWinWin system, has shown that the wiki technology can be used to effectively facilitate project stakeholders in sharing information, resolving issues, and eventually reaching mutually satisfactory requirements. Further experimental evaluation was conducted with more student projects (20 project teams). The initial results showed that the teams with at least one member acting as an active (wiki) shaper were more likely to achieve better outcomes in requirements negotiation (Wu et al., 2009). Similarly, Ferrerira and Silva (2009) designed an enhanced wiki system that provided a collaborative environment to support requirements specification with a controlled natural language, and to support requirements negotiation and management. However, the system has not been evaluated.

Very few research studies have been done on different uses of wikis in software design and development projects. One study evaluated wikis as a communication tool for collocated and distributed teams, and discovered that collocated teams used the wiki to document, but not as a collaborative working space while distributed teams used the wiki for both collaboration and documentation (Walthall et al., 2011). Another study utilized wiki to create a collaborative programming environment for software engineers (Hattori, 2011). Wikigramming was developed and implemented. The study found that Wikigramming was suitable for a training environment for novice programmers because new programmers could quickly learn programming by copying, changing, and correcting other programmers' code.

2.3.4 Wiki research gaps

Although wiki research has started since 2001, the number of empirical studies on the use of wikis is still relatively low. When considering the context of use, the majority of the studies fall into public collaborative web authoring (e.g. Wikipedia), educational use, and organizational knowledge management (e.g. intranet and knowledge base). Interestingly, wikis are often used by design and development teams (Poole & Grudin, 2010; Majchrzak et al., 2006), but very few studies have examined the use of wikis in the design context. The reviewed studies and a few other studies related to the use of wikis in software design and development are more system-development focused, and have focused on particular aspects of design collaboration. This dissertation, in contrast, investigates the use and user experience, focusing on the information sharing which is an integral part of design collaboration that spans across the project lifecycle. It aims to provide empirical evidence on how wikis are used in interdisciplinary design, more specifically, how wikis are used to support information sharing across disciplinary groups in interdisciplinary design projects.

CHAPTER 3: CONCEPTUAL FRAMEWORKS

To understand how interdisciplinary software design teams use wikis to share information – what and how they actually share information – and to be able to best utilize wikis to support information sharing, I drew on the 3-T framework (Carlile, 2004) and Cognitive Work Analysis (Vicente, 1999). According to Vicente (1999), there are three generic approaches to work analysis. Those approaches are normative, descriptive, and formative. The normative approach aims at prescribing how a system *should* behave. The descriptive approach focuses on describing how a system currently behaves. The formative approach focuses on identifying how the system *could* behave under given constraints. Based on this description, the 3-T framework is considered a descriptive approach that I used to conceptualize various information sharing activities in design projects. On the other hand, the Cognitive Work Analysis is considered a formative approach that I used to understand how wikis could be used to support information sharing in interdisciplinary design. In this chapter, I describe each framework and their application in the study.

3.1 The 3-T Framework

The notion of boundary appears in several important concepts used to study information-sharing related phenomenon, for example, boundary objects, boundary spanning, and knowledge boundary. It is implied that a boundary exists in the interaction between two or more groups, for examples, social worlds (Star & Griesemer, 1989) and object worlds (Bucciarelli, 1994). Since this dissertation investigates information sharing across disciplines, a boundary exists in information sharing between the actors (a sharer and a receiver) who belong to different disciplinary groups. Thus, the theoretical basis for this research draws on the concept of boundary spanning, which aims to explain communication and information processing across the boundaries between two or more groups (Carlile, 2002; Gasson, 2005; Sonnenwald & Lievrouw, 1997). Carlile's integrative framework for managing knowledge across boundaries (Carlile, 2004) promises to be a particularly effective framework because it focuses on the effectiveness of sharing and assessing knowledge across boundaries between domains of knowledge and experience.

Carlile (2004) developed an integrated framework for managing knowledge across boundaries (3-T framework) to describe progressively complex boundaries and processes that occurred during knowledge sharing across functional groups. The framework combines three different approaches to moving knowledge across functional boundaries in the product development. The three approaches are syntactic, semantic, and pragmatic. Based on Shannon and Weaver's mathematical theory of communication and information processing (Shannon & Weaver, 1949), a syntactic approach establishes a shared and stable syntax or language to ensure a quality information exchange. A semantic approach, however, recognizes that a shared syntax alone is not sufficient because interpretations can be different because individuals often have different meanings in their functional group. Therefore, to move knowledge across boundaries, the differences in interpretations must be taken into account. A pragmatic approach, proposed by Carlile (2002) to complement the first two approaches, recognizes that to successfully move knowledge across boundaries, individuals must be willing to assess new knowledge as well as to change their own knowledge. This process is called "knowledge transformation" (Carlile, 2002, p. 445).

Integrating all three approaches, the 3-T framework describes that there are three levels of boundaries (syntactic, semantic, and pragmatic) and associated processes (transfer, and translation, and transformation), and that moving knowledge across each type of boundary requires a different process (shown in Figure 3-1). At a syntactic boundary, transferring knowledge with a common syntax or language is sufficient. To cross a semantic boundary, it requires more than just a common syntax; common understanding needs to be developed through knowledge translation. At a pragmatic boundary, knowledge needs to be transformed to result in altering current knowledge, creating and validating new knowledge collectively.

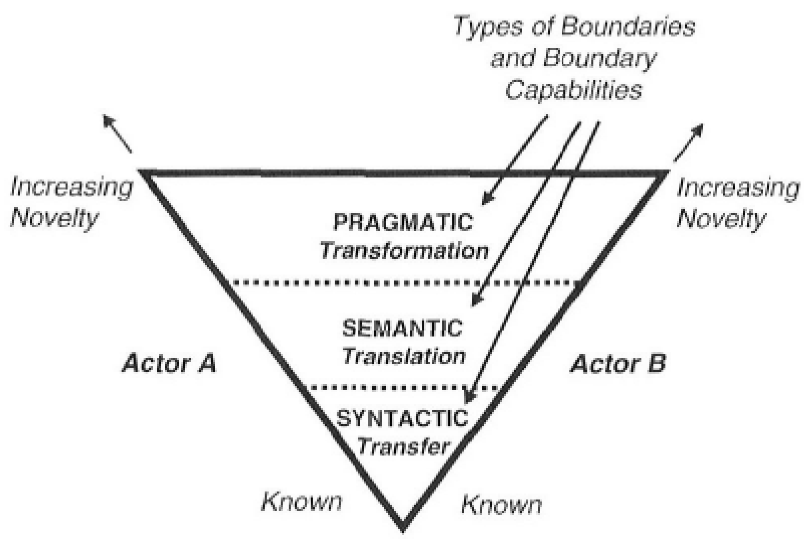


Figure 3-1: 3-T framework (Carlile, 2004)

Based on its characteristics and use, the 3-T framework is considered a descriptive framework, which focuses on how knowledge moves across boundaries. To illustrate how the 3-T framework has been used to describe knowledge management across boundaries, I will use a scenario from a new product design and development project, which was used as an example in Carlile (2002). In the scenario, the project's goal was to design and develop a new safety and environmental valve for automobile fuel system. The project required collaboration among different functional groups including sales, design engineering, manufacturing engineering, and production. In a design review meeting, Mick (a manufacturing engineer) informed the design team how a current design would create assembly and testing problems, and proposed a new design as a solution. While Mick was sharing his knowledge to other functional groups, the team was engaged in three complex processes as depicted in the 3-T framework. First, to demonstrate his point, Mick used an assembly drawing, which shows a three-dimensional view of product parts along with measurement. At the syntactic boundary, the assembly drawing allows Mick to represent and share his knowledge because it contains a shared symbol or language among the team members. Next, at the semantic boundary, the, other team members must learn and come to a mutual understanding of the problem that Mick raised. Finally, to cross a pragmatic boundary, all functional groups had to transform and integrate Mick's knowledge into their own current knowledge to solve the problem and evaluate Mick's proposed solution. For example, a design engineer had to understand the problem and modified the design specification that would solve the problem.

The dissertation uses the 3-T framework to examine the effectiveness of wikis in supporting sharing design-relevant information (e.g., design documents, meeting minutes, sketches, and diagrams) across disciplinary groups in design teams. The 3-T framework is an appropriate framework in investigating information sharing because of not only its boundary-spanning aspect, but also its recognition of multiple sharing processes.

3.2 Cognitive Work Analysis

Given my interest in exploring how wikis could be implemented to better support information sharing in design projects, I need a formative approach that not only helps me understand information sharing practices and the use of wikis in the design situation, but also helps me develop design recommendations for wikis. By design recommendations, I refer to not only technical design, but also social, organizational and work practices that will promote effective use of wikis during the design processes. The 3-T framework is considered a descriptive approach, which explains current information sharing processes but it does not answer the question, "how could wikis be implemented to better support information sharing?" In addition, I argue that information sharing cannot be understood by examining only the sharing activities alone. It requires a holistic perspective to understand how other factors may shape the sharing process and outcomes. Therefore, I chose Cognitive Work Analysis (CWA), a formative approach, to complement the 3-T framework, a descriptive approach, in order to examine the use of wikis for information sharing in a design context, and in order to translate findings into design recommendations.

Developed by Rasmussen, Pejtersen, and Goodstein (1994), CWA is a formative work analysis framework, which aims at specifying the requirements of what *could be* (Vicente, 1999). It focuses on identifying and analyzing the factors, namely constraints and goals, which shape information behavior and work practices. CWA allows researchers to investigate a subject of interests from multiple dimensions – environment, work-domain, organizational, activity, and individual – simultaneously. CWA dimensions are shown in Figure 3-2. Each dimension represents where constraints come from.

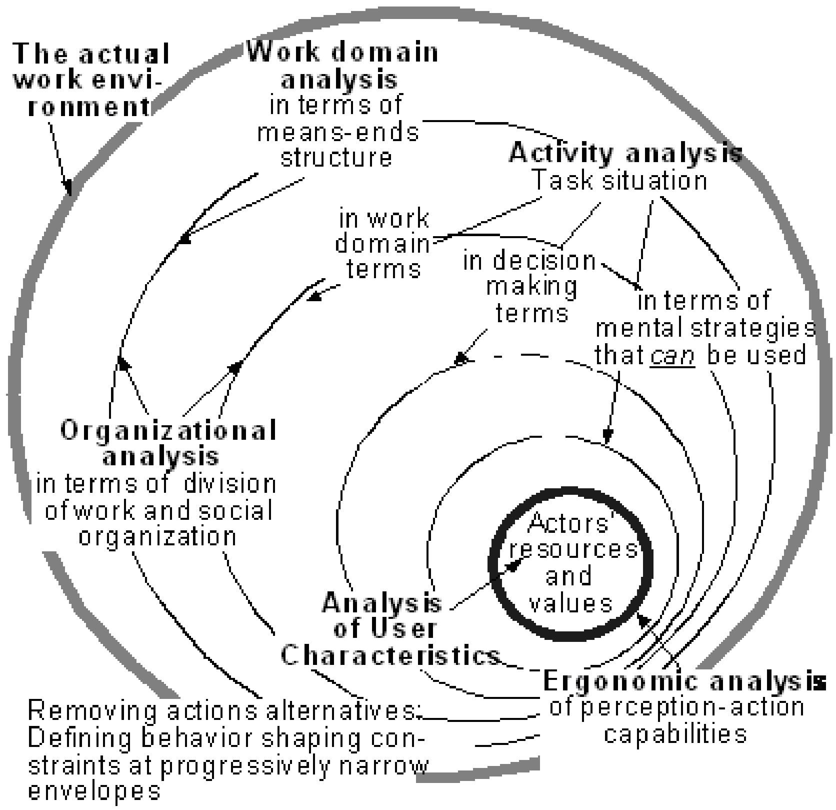


Figure 3-2: The dimensions of Cognitive Work Analysis (Fidel & Pejtersen, 2004)

CWA dimensions:

1. *The actual work environment* is the external environment of the system that affects how the system is operating. The work environment helps identify the goals and the constraints of the systems; it answers the question why the system exists. However it is not controllable or changeable by the actors.

2. *The work domain* refers to the system itself, which imposes constraints on how actors carry out tasks.

3. *The task situation* (activity analysis) represents a set of tasks that need to be accomplished. That is, it focuses on what needs to be done and how they can be done.

4. *Social organization* deals with the relationships between actors, and between actors and the organization, such as, how the work is allocated among actors, how actors are organized into groups or teams, how actors communicate and cooperate with each other.
5. *Actors' resources and values* represent a set of constraints associated with the prototypical actors themselves, for example, their skills, knowledge, and experience. The prototypical actor is the best example of an actor in the studied context (Fidel & Pejtersen, 2004).

CWA has been used successfully to study information behavior, particularly for the purpose for information system design (Fidel & Pejtersen, 2004). The Design Explorer Project (Pejtersen et al., 1997) is an example of successful application of CWA to study information sharing in design. Information exploration needs were identified and analyzed according to multiple dimensions, including task situations, decision strategies, characteristics of different design participants, and organizational context and communication roles. The analysis was then transferred into a set of requirements for an information system that could be used to support information exploration in design. Another example is a study of collaborative information retrieval and information sharing in design teams by Poltrock et al. (2003). Guided by CWA, Poltrock and colleagues conducted field studies of information gathering in two design teams, a software design team and a hardware design team. This study not only discovered information behavioral patterns but also offered suggestions on how collaborative technologies could better support collaborative information retrieval.

CWA is an effective framework to study a "complex system;" it provides concepts and tools (discussed in section 3.3 below) to facilitate an analysis of such complex system without reducing its complexity. Since design is complex, CWA is suitable to study the use of wikis to support information sharing in design. I expect CWA to help me understand the context and its constraints that shape how design teams use wikis to share information, and in turns generate recommendations for wiki design that can support information sharing.

3.3 Applying the Two Frameworks

> *During a conceptual design phase, a designer, John, asked a developer, Ed, for statistical data about the tagging feature of an existing tool. Ed posted the data on the project wiki, and told John on an IRC channel that the requested information was ready. John looked at the data on the project wiki and had a few questions. So, he had a face-to-face conversation with Ed to clarify his understanding. Once John understood all the statistical data provided by Ed, he had a better understanding of how people were using the tagging feature. John could then make a design decision about the feature.*

The short story above was created based on my observation of a software design project. It is presented to illustrate how I applied the two frameworks in the design context. To understand the above information sharing between John and Ed, both the CWA framework and the 3-T framework were used to develop specific research questions, guide data collection, and frame data analysis. The 3-T framework is used as a means to conceptualize information sharing during design processes, while CWA framework brings into account factors that affect information sharing in the studied context, software design and development. Based on the literature review, I adapted the 3-T framework to account for sharing physical artifacts. The adapted framework, 4-T framework, is illustrated in Figure 3-3.

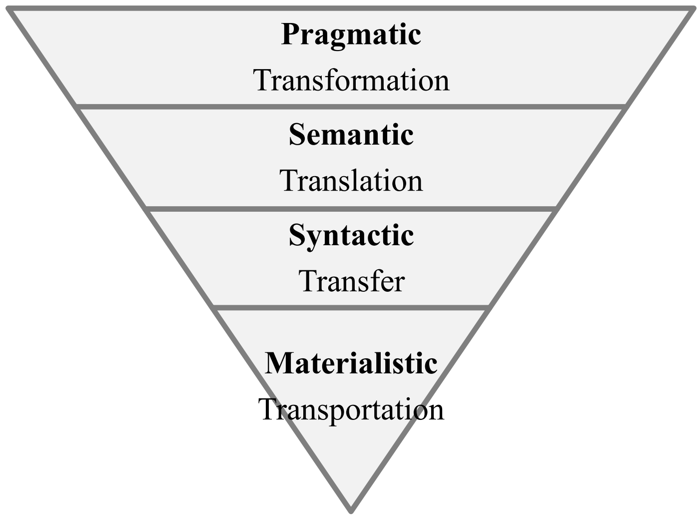


Figure 3-3: 4-T framework (adapted from the 3-T framework)

To fully accommodate all aspects of information sharing, I added another level to the framework, materialistic sharing and transportation process, to account for sharing tangible objects, which was not addressed by the 3-T framework. The 4-T framework is described below:

> *Materialistic sharing* results from a successful *transportation* of information forms. Transportation refers to a process by which an information artifact is shared. At this point, the actors are concerned with how to move an information medium (e.g., electronic files and documents) between them. Therefore, the sharing is successful when the receiver has access to or gets an object that contains the information.

> *Syntactic sharing* results from a successful *transfer* of information. Transfer refers to a process by which the information content (a message) is shared. This process requires a common syntax or language. The aim of transferring information is to get a message across from the sharer to the receiver. Therefore, the sharing is successful when the receiver gets the information in a form that is comprehensible.

Semantic sharing results from a successful *translation* of information. Translation refers to a process of establishing a common understanding. The actors need to ensure that the meaning of information is shared. Therefore, the sharing is successful when the receiver not only gets the message contains in information mediums, but also the meaning that the sharer intends to share.

Pragmatic sharing results form a successful *transformation* of information. Transformation refers to a process by which shared information is utilized to generate new information or to carry out some kinds of work. The actors may negotiate with each other during the transformation process. Therefore, the sharing is successful when the receiver is able to utilize the shared information in some effective ways.

Following the CWA approach, I started by constructing the context of the current information sharing practices and the way in which software design teams used wikis, and, based on this description, developed recommendations for the future design of wikis and for the effective use of wikis. The 3-T framework complemented nicely to the "Activity Analysis" layer of the CWA by offering a distinct perspective on boundary and multi-level process of information sharing activity. In addition to decision-making and strategies, information sharing activities were analyzed in terms of 3-T framework. That is, sharing processes and outcomes involved in information sharing activities were identified. Identifying CWA dimensions is the first step to dealing with the complexity from the complex design environment imposed on information sharing practices and the use of wikis. The CWA dimensions for the analysis, incorporating 3-T framework, is described as follows:

- The work environment: The environments in which the design project is operated – e.g., corporation or organization, regulations, design standards, stakeholders.
- Work-domain analysis: The work that is done in the design project using means-ends analysis (described in detail in chapter 4).
- Organizational analysis: The collaboration, the management style, the organizational culture, the social conventions, and role allocation.
- Task analysis: Specific tasks at three levels – information sharing tasks, and wiki tasks – that are carried out by the design teams using means-ends analysis (described in detail in chapter 4).
- Decision analysis: A more specific analysis of individual decisions when sharing information through wiki.
- Strategies analysis: Information sharing strategies possible to support wiki tasks.
- User's resources and values analysis: prototypical characteristics for each discipline group of team members – e.g., experience, knowledge, most important values. This dimension also complements the pragmatic boundary of the 4-T framework as user's resources and values directly affect the transformation of knowledge.

The 4-T framework and the CWA framework together guided the data collection and data analysis with the aim of answering my research questions. The CWA framework helped address the first research question, *"How do design teams share design-relevant information across disciplinary boundaries?"* With the CWA framework, I was able to understand both the context and the constraints of information sharing practices of interdisciplinary design teams during the software design and development process. Then, the 4-T framework and the CWA framework were used together to answer the second research question, *"To what extent do wikis support information sharing and interdisciplinary design activities?"* The 4-T framework addressed how effective wikis could support each level of information sharing, while the CWA framework addressed multiple factors such as the work environment and design tasks that may affect information sharing and the use of wikis. Finally, the CWA framework addressed directly to the last research question, "How could wikis be enhanced to better support information sharing in interdisciplinary design?" because the formative approach provided by the CWA framework enabled me to transform the research findings to system requirements for the future design of wikis and for the effective use of wikis.

CHAPTER 4: METHODOLOGY

The objectives of the dissertation are to: (1) Study how interdisciplinary design teams use wikis; (2) Identify best practices for the effective use of wikis; and (3) Examine how wikis might be enhanced to better support information sharing in design projects. To address these aims, I investigated how software design teams used wikis to share information across disciplines. This chapter presents research questions, describes research sites, and discusses research methodology including data collection and data analysis. The credibility and validity of this research is also addressed.

4.1 Research Questions

With this dissertation, I answered the following research questions:

How do design teams share design-relevant information across disciplinary boundaries?

1. What information is shared across disciplines?

2. How do design teams share project information across disciplines?

3. What information technologies do design teams use for sharing information?

To what extent do wikis support information sharing and interdisciplinary design activities?

4. How do design teams choose to employ wikis?

5. What design activities are supported by the use of wikis?

How could wikis be enhanced and utilized to better support information sharing in interdisciplinary design?

6. What features of wikis facilitate information sharing among design team members?

7. How do the work environment, work domain, organizational and social structure, tasks, and users' value and resources influence the use of wiki in design team?

8. How could wiki use be improved for interdisciplinary design collaboration?

To answer the research questions, I conducted an exploratory study of five interdisciplinary design teams in three organizations. At the time of the fieldwork these design teams were using wikis as their central information sharing and management system. An investigation of multiple design teams provides a better understanding of the use of wikis, and assures that the findings are not idiosyncratic to a particular team. Data collection and analysis was guided by both the 4-T framework and the CWA framework. While the 4-T framework helps explain how effective wikis are in supporting information sharing, CWA particularly informs how wikis could be implemented to support information sharing in interdisciplinary design.

4.2 Data Collection Methods

I collected both qualitative and quantitative data using multi-method data collection strategy in order to provide rich data about the use of wikis. Methods included observations, interviews, and wiki review. Observations were conducted to understand information sharing practices and how wikis fitted into the design process, which would have been difficult to capture solely by interviews. The objective of interviews was to gain a better understanding on how and why individual design team members used wikis from their own perspectives. Lastly, I reviewed wiki sites and collected wiki log files, to gain additional insight on wiki usage, including types of design-relevant information shared through wikis, levels of information sharing, and levels of participation of team members. In this section, I describe each data collection method in detail.

4.2.1 Observation

Observation is a qualitative research method that enables the researcher to fully understand and capture the context of interest, which is essential to develop a holistic understanding of the phenomena under study (Patton, 2002). Observation methods are often categorized into two types based on the researcher's involvement – participant observation and non-participant observation. Participant observation involves the researcher observing participants in their daily lives and participating in their activities as a participant for an extended period of time. In contrast, when conducting a non-participant observation, the researcher generally remains apart from the participants being studied and does not engage in their daily activities. For this dissertation, both types of observation were employed.

The objective of the participant observation was to gain an in-depth understanding of the design context, information sharing practices, and the use of wikis throughout the design process. The purpose of participant observation was to document in detail how team members shared information and used a project wiki site throughout a design process. Participant observation was an appropriate method because it provided me a tacit understanding of information sharing and the use of wikis. From my experience working on design projects, information-sharing activities sometimes occurred while team members were interacting in meetings or performing tasks, which could be difficult to articulate in the interviews or observed in wikis. Field notes were taken during project meetings, group activities, and at the end of each day. The field notes included both what actually happened from direct observation and my reflections on what I observed.

For the project teams that I was not a participant observer, I observed project meetings at least once. The objective of the project meeting observation was to gain a better understanding of nature of the project, language used by the teams, social organization, and team dynamic. The observation was also to help me establish rapport with participants whom I interviewed later on. During a meeting, I paid special attention to information sharing activities and any references or use of a project wiki site.

4.2.2 Interview

The objective of the interviews was to gain a better understanding for how and why design teams use wikis. The data from the interviews were used to identify themes or patterns in (1) usage, (2) expectations, (3) experience, and (4) success criteria. The interviews focused on team members' roles, tasks, information sharing practices, expectation for wikis, and experience using project wikis.

Not every individual on each design project was interviewed. The choice of individuals was based on their roles, wiki participation, and availability. For each team, I interviewed a team leader, at least one person from each disciplinary group, and members with highest and lowest wiki participation. Each interview took anywhere from 30 to 60 minutes. The first half of interview focused on participants' roles, tasks, their information sharing practices, and what they hoped to achieve when using project wikis. The second half focused on their experience and their perspectives on the project wiki. In general, most interviews were conducted towards the end of project so that participants would have experience with the project wiki. All interviews were in-person, audio recorded, and transcribed. The interviewed participants signed a consent form (Appendix F) before participating in an interview.

An interview guide (Appendix A) was used in all interviews. Interview questions were, for example, "What information do you put on the project wiki?", "How often do you access the project wiki?", "How did you make a decision what to post on the wiki?", "What do you like about the wiki?", "Overall, how do you describe your experience using the project wiki?" When possible, I also asked participants to go through a project wiki, and explain whether or not they accessed and used information items (articles) on the wiki. If so, what did they use them for?

4.2.3 Project wikis review

In addition to observation and interviews, I reviewed the project wiki sites, both organization and content, to examine what and how design-relevant information was shared through the project wikis. The objective of wiki review was to provide additional insight into the activities on project wiki sites, particularly who, what, and when aspects of the activities. It was a way to triangulate the data about the use of wikis that was obtained from the interviews, "what people said they did" vs. "what they actually did." The wiki sites of completed projects were reviewed. The project wiki log files were also collected for quantitative data analysis on wiki usage such as size, contribution patterns, and feature usage. A wiki log file is a server log automatically created and maintained by the server of activity performed on the wiki sites. Information contained in the wiki log file includes, for example, a history of page edits, a number of page views, and page size.

4.3 Research Sites

The study employed a purposeful sampling technique. Several organizations that potentially used wikis in their design and development projects were contacted. Drawing on personal connections, I sent an email message, in which I explained my research interests and asked if they would be interested in participating in the study. Once the contact persons expressed an interest, I worked with them to identify design teams to recruit for the study.

The following inclusion criteria were used to recruit design teams; each team:

- Used wikis as a fundamental information-sharing tool.
- Was an interdisciplinary team consisting of members from at least 3 disciplines (functional groups).
- Was a small-size team with 5-12 members where the majority of team members were collocated.

Through this recruitment process, I successfully recruited five project teams that met the inclusion criteria from three organizations[1], as shown in Table 4-1.

[1] Pseudonyms are used to protect confidentiality and privacy of participating organizations, teams, and participants.

Table 4-1: Participating project teams

Project	Organization	Product being designed	Project duration	Project Size	Disciplines represented
A	State University	Web-based collaborative space	13 months	9	Management, user research, interaction design, software engineering
B	State University	Online discussion board	6 months	6	Management, interaction design, software engineering
C	Flynow	Automated ticket reprice/reissue system	12 months	11	Management, marketing, interaction design, software engineering
D	Flynow	Online voice-enabled virtual assistant application	9 months	9	Management, marketing, software engineering, customer care
E	Digital Media	Global search system	Ongoing at the time of study	11	Management, user research, interaction design, software engineering, linguistic

All the project teams engaged in software design and development projects in very different industries. Two teams were part of a large state university (25,000 employees and 50,000 students), two teams were part of a major air carrier (10,000 employees), and one team was part of a digital media company (2,000 employees). At State University, I collected data through participant observations, interviews, document review, wiki site review, and wiki log file. For other two organizations, Flynow and Digital Media, I collected data through interviews, non-participant observations, and wiki site review. During two years of data collection (April 2007 to March 2009), I conducted 9 months of participant observation (at EdApp, State University, only), observed over 60 project meetings, conducted 33 interviews, reviewed 4 project wiki sites and 3 organizational wiki sites, and collected one wiki log file (which contained data from wiki sites of multiple projects). The collected data are summarized in Table 4-2. Research sites and data collection at each site are described in more detail, as follows.

Table 4-2: Data collection summary

Research sites	Interviews	Observations	Wiki review	Log file
State University				
Project A	6	52	1	1
Project B	6	2	1	1
Others (past projects)	N/A	N/A	2	N/A
Flynow				
Project C	4	1	1	No
Project D	7	1	1	No
Others (project managers)	2	N/A	N/A	N/A
Digital Media				
Project E	5	1	1	No
Others	3	N/A	N/A	N/A
Total	33	57	7	2

4.3.1 State University

As I started looking for potential study sites for my dissertation research, I approached EdApp, a software development group in State University, a research university, because the group just started using wikis in their software development projects. In addition, I had worked at EdApp as a part-time user experience designer since 2005, and had been involved in 12 software development projects. I had been involved in various stages of the design process, including requirement gathering, conceptual design (developing use-case scenarios, persona, and task flow diagram), user interface and interaction design, and usability testing.

Team A was working on a web-based workspace project. The project goal was to design and develop a web-based collaborative workspace application that can be used for classes or projects. The project took approximately one year to complete (February 2007 to March 2008). The team consisted of a project manager, two software developers, three interaction designers (two full-time and one part-time), two researchers, and a QA engineer. Of 9 members on this team, 7 members had prior experience using wikis for previous projects.

Team B was working on a major release project for an online discussion board. The project goal was to design and develop additional features for an existing online discussion board to meet the needs of students, faculty, and staff. The project took approximately six months (May 2007 to October 2007). Team B consisted of a project manager, two software developers, two interaction designers, and a QA engineer. All team members had experience using wikis in previous projects.

Data collection at State University started in April 2007 after I received a human subjects approval from the Internal Review Board, and continued until April 2008 as shown in Table 4-3.

Table 4-3: Data collection at State University

Time period	**Data collection activity**
April 2007 – March 2008	Participant observation on Project A
June – September 2007	Interview Project Team B
January – March 2008	Wiki review
March – April 2008	Interview Project Team A

Because of my role as a part-time user experience designer, my researcher role became a participant observer when collecting data at EdApp. During that time, the group was working on two major projects – Project A and Project B. As a part-time interaction designer, the level of my involvement, however, was different in Project A than Project B. I was a member of Project A team, taking an interaction designer role for the project, while I was not a project member on the Project B. Although I was not a member of Team B, I was still involved in its design review process, attending many design critique meetings.

Over the course of 9 months, I participated in over 50 meetings for Project A and Project B. As an interaction designer, I participated in most of Team A meetings (formal and informal), and several design critique meetings of Team B. To reduce observer effect, I recorded field notes in a notebook that I already used and carried around for my design job, and then I transcribed my notes into my research journal at the end of day. The observation progressed from unfocused observation to focused observation. Unfocused observation, also called descriptive observation, is conducted when the researcher tries to observe and record as much as possible without having a particular question in mind, whereas focused observation is conducted when the researcher tries to answer more specific questions (Spradley, 1980). Since I have been in the context for so long, unfocused observation technique allowed me to pay attention to everything, including things that may seem obvious, and to reduce an influence of familiarity with the context (Jorgensen, 1989). Therefore, in the first few months, I took notes on every aspect of the meetings, including attendees, meeting space layout, meeting goals, meeting structure, information sharing (what, why, and how), and participants' use of wikis. After I had a clear idea of how meetings were structured, my attention was directed toward information sharing activities, in which teams made use of or referred to wikis.

I had interacted with all members from both teams during the time of study, and conducted a formal interview with 12 team members (6 members from each team). Team B members were interviewed in September 2007 (except for the project manager whom I interviewed in June 2007 because she was taking a leave after that), while Team A members were interviewed around April 2008. The decision to interview toward the end of projects was made to reduce any influence of the study on their use of wikis.

Throughout the data collection period, I occasionally reviewed project wiki sites and project documents. Once both teams completed the projects, I collected a wiki log file, which contained log entries for multiple wiki sites used in multiple design and development projects.

4.3.2 Flynow

Flynow is an air carrier that has over 10,000 employees. With its spirit of technical innovation, the company often has ongoing IT system design and development projects, which are managed through the Information and Technology department. At the time of study, Flynow used Openwiki (www.openwiki.com), a text-based wiki, for their design and development projects.

Team C was designing and developing an automated ticket reprice/reissue feature to facilitate travel agents and customers in changing airline tickets. The designed system was called was "Ticket Reprice and Reissue System (TRRS)." The team consisted of a project manager, a product manager, three QA systems analysts, four software developers, an interaction designer, and a pricing analyst.

Team D was designing and developing a virtual assistant application, for the company website. Based on natural language questions, the application would provide a brief answer and navigate to the page on our site with the most relevant information to website customers. This project was Flynow's latest customer-friendly innovation. The goal was to improve customer self-service and to reduce incoming calls to the call center. The team consisted of a project manager, three representatives from e-commerce, three employee-facing developers, three customer-facing developers, a representative from a company website service, a customer loyalty staff, a QA engineer, and an e-business staff.

Data collection at Flynow took place from April 2007 to July 2008 as shown in Table 4-4. Initially, I had planned to observe several team meetings and interview members for all participating teams. However, I was given an opportunity to observe only one team meeting, which was a post-release meeting of TRRS team. At the beginning of the meeting, I introduced myself as a graduate student doing research on the use of wikis by software development team. There were 7 project members attending the meeting. It was my first time meeting the team; therefore, my attention was on getting to know the team members as well as information sharing activities occurring during the meeting.

Table 4-4: Data collection at Flynow

Time period	Data collection activity
April-May 2007	Team C (TRRS) meeting observation Wiki review
September 2007	Interviews both project teams Wiki review
June – July 2008	Interviews after projects were completed

I scheduled interviews with team members via a contact person, and conducted a total of 13 interviews with 10 participants, who represented six functional groups – Project management, design, eCommerce, customer care, customer facing development, and marketing. Three participants were available for a follow-up interview after the projects were completed while others were interviewed only once in the middle of the project.

Wiki review happened during some of my site visits when I was granted access to the company wiki sites. I captured screenshots of participating team wiki sites, and took notes on their usage. In addition, a few participants gave me a print out of wiki pages that they frequently visited. I was not able to collect a wiki log file for reasons of corporate confidentiality.

4.3.3 Digital Media

Digital Media is a leading creator and distributor of images, footage, music and other digital media content, providing the world's largest imagery collections. It is a global company with approximately 2,000 employees with twenty offices worldwide. At the time of study, different wiki platforms were used by different functional groups with an effort to reorganize and integrate different wikis into one central wiki to be used company-wide.

Team E was a customer web experience team. At the beginning of the study, the team was involved in the online music search project, which integrated an online music search and sale system into the company website. A few months later, the team worked on a global search system because the online music search project was canceled. The team consisted of a project manager (also a product owner), a scrum master (also a software developer), 4 software developers, and 3 QA engineers.

Data collection at Digital Media spanned over a year as summarized in Table 4-5. I first approached Digital Media through its IT director in September 2007 and had a formal discussion with him in February 2008. In the first meeting, I learned that wikis had been widely used at Digital Media, so I sought to get an approval to recruit one or two software development teams for my dissertation.

Table 4-5: Data collection at Digital Media

Time period	Data collection activity
February 2008	A meeting with the IT director and tour of the wikis
April 2008	Interviews Wiki review
July-August 2008	Project meeting observation Interviews
March 2009	Interviews Wiki review
June 2009	Interviews

In summary, I observed one project meeting and interviewed 6 participants from 4 functional groups – project management and organization, quality assurance, software development, and network operations center. Two participants were involved in an effort to restructure multiple wiki sites used at Digital Media, which I had an opportunity to discuss during interviews. I reviewed the content of a project wiki site during my site visits and as time permitted and took notes on the types of information the Project Team E shared on their wiki site.

4.4 Data Analysis

The unit of analysis in this dissertation is the individual design project team because my main interest was on how each team used wikis to support information sharing during the design process. Data analysis presented a challenge because not only were many data types involved, but also different data types were available from each study site as previously shown in Table 4-4. There are four major types of data collected in this dissertation – field notes from observations, interview transcripts, wiki content, and wiki log data.

While research frameworks often instruct researchers on what methods to use or what questions to ask, CWA does not. Instead, it offers a general approach that requires the individual researcher to select the appropriate methods and the specific questions to ask, based on the phenomenon that is being investigated (Fidel & Pejtersen, 2004). As a methodological framework, CWA provides a general structure and tools for analysis. The structure identifies the major areas that should be investigated. These are called *dimensions of analysis* namely work environment, work domain, social organization, activity (task, decision, and strategy), and actor's resource and values. These dimensions represent the constraints that shape the information sharing practices and the wiki usage of design project members. I applied various analysis tools, including a means-ends analysis template, information flow diagram, thematic approach, and quantitative analysis of log data. I used Atlas Ti for qualitative data analysis, and Microsoft Excel and Microsoft Access database for wiki log data analysis. Table 4-6 illustrates how I conducted each dimension of analysis for each organization and each design project.

Table 4-6: Data analysis method summary

Analysis Dimension	Methods and Tools	Data sources
Work environment	Interpretive analysis	Observation notes, interview scripts, company websites
Work domain	Means-ends analysis template, Atlas Ti	Observation notes, interview scripts, wiki content
Social organization	Thematic analysis, Atlas Ti	Observation notes, interview scripts
Activity: Information sharing	Means-ends analysis template, thematic analysis, Atlas Ti, information flow diagram	Task observation notes, interview scripts, wiki content
Activity: Wiki task	Inductive thematic analysis, Excel, Access database	Wiki content, wiki log, interview scripts
Activity: Wiki decision	Inductive thematic analysis	Interview scripts, wiki content
Activity Wiki strategies	Inductive thematic analysis	Wiki content, interview scripts
Actor's resources and values	Thematic analysis, Excel, Access database	Interview scripts, wiki log

4.4.1 Work environment

Each organization represented a work environment for its project teams. To understand three work environments, one of each organization, I employed an interpretive analysis method to analyze observation notes, interview scripts, and company websites. The results of the work environment analysis were used to write narratives for organizations and organizational units that participating project teams belonged to. The narratives describe elements outside of the projects that could affect the projects, which include organizations, industry, size, organizational structure, and physical layouts.

4.4.2 Work domain

The work domains in my analysis are five design projects, in which participants used wikis. I applied the means-ends analysis template (Figure 4-1) to define levels of abstraction for each design project. The means-ends analysis made it possible to analyze the system without removing the complexity inherent in the context of study (Vicente, 1999). The aim of the analysis was to answer three goal-oriented questions – Why? What? and How? Through means-ends analysis, I was able to identify the projects' goals, priorities, constraints, general work functions, general work processes, and physical resources. Physical resources represent the most concrete level of the analysis, while goals and constraints represent the most abstract level. Moving up in the hierarchy is answering the "why" question while moving down the hierarchy is answering the "how" question.

Goals **Constraints**	What are the ultimate goals and purposes for the project? What affects the project activities but cannot be changed?
Priorities	What can be done to achieve the project goals given the constraints?
General work functions	What are the general project functions?
General work processes	What actual activities take place?
Physical resources	What tools and resources are being used to perform tasks?

Figure 4-1: Means-ends analysis template for projects

4.4.3 Social organization

Social organization analysis aims at understanding work allocation (that is, how tasks were divided), the nature of project organization, team culture, and collaboration patterns. Based on the observation notes and interview transcripts, I describe the social organization in terms of project teams' structure and nature of collaboration.

4.4.4 Activity

Information sharing activity: To analyze information sharing activities, I started with means-ends analysis for each project based on observation notes and interview scripts. The means-ends analysis template for information sharing is shown in Figure 4-2. Using the results of means-ends analysis, along with observation notes, I created information flow diagrams for shared information items identified in the means-ends analysis to capture information input and output of each disciplinary group and how information is moved between groups. The diagram was a graphical representation of the flow of information of each design project. It was adapted from the communication flow model (Beyer & Holtzblatt, 1998) and the information flow map (Davis et al., 2001). I applied the 4-T framework into the diagrams in order to code different sharing processes, namely transport, transfer, translate, and transform.

Goals **Constraints**	What are the ultimate goals and purposes for sharing information across disciplines? What affects the information sharing activities but cannot be changed?
Priorities	What can be done to achieve the information sharing goals given the constraints?
General work functions **(4T)**	What is done in general terms? Materialistic sharing Syntactic sharing Semantic sharing Pragmatic sharing
General work processes	What actual sharing activities take place?
Physical resources	What information is being shared? (and information items) What is being used to perform the information sharing activities? (Information sharing tools)

Figure 4-2: Means-ends analysis template for information sharing

Using the results of means-ends analysis and information flow diagrams, I answered the first research question, "How do design teams share design-relevant information across disciplinary boundaries?" More specifically, I discussed what, why, and how information was shared across disciplinary groups during the design process.

Wiki activity: To understand how design teams used wikis to support information sharing, I conducted the analysis on three layers – wiki task, wiki decision, and wiki strategies. The wiki content and log files were the primary data source for the wiki activity analysis. The results of wiki activity analysis were used to answer the second research question, "To what extent do wikis support information sharing and interdisciplinary design activities?"

For wiki task analysis, I reviewed the wiki content and page history to identify wiki processes. The wiki processes were coded into information sharing processes identified through information sharing activity analysis. Interview data were used to augment the coded wiki content and log data.

The goal of wiki decision analysis was to understand what decisions design teams had to make when sharing information via wikis. I applied the deductive thematic analysis approach to analyze the interview data and wiki content in order to find themes on wiki decisions.

The goal of wiki strategies analysis was to identify categories of wiki processes that design teams performed to accomplish information sharing via wikis. I applied the inductive thematic analysis, with 4-T framework, to further analyze the findings from the wiki task analysis in order to describe wiki strategies employed by design teams.

4.4.5 Actors' resources and value

The analysis of actors was conducted to understand how design team members' resources and values could influence the way in which they used wikis to share information across disciplines. Interview data and wiki log data were analyzed using a thematic analysis approach to identify team members' experiences and skills, wiki contributions, perceived value, and attitudes toward wikis.

CHAPTER 5: UNDERSTANDING THE CONTEXT

This chapter presents the context of wiki use by interdisciplinary design teams. In the first section of this chapter, I describe the individual research sites through narratives, constructed from observation, interview, and wiki review data. The individual research sites are three organizations and their participating design projects: State University (Project A and Project B), Flynow (Project C and Project D), and Digital Media (Project E). These research sites represent individual contexts, in which the use of wiki technology by interdisciplinary design teams was situated. In the second section, I generalize the individual sites of the three organizations and five design projects as a single collective context, which represents a common setting of wiki use by interdisciplinary design teams. I do so by discussing these research sites in terms of CWA analysis dimensions in order to highlight similarities and differences. The major goal of this chapter is to provide a holistic view of the context of wiki use investigated in this dissertation, whereas in the next chapter I will discuss how the software design and development teams used project wiki sites to support information sharing.

5.1 Individual Contexts

The three narratives aim to describe the individual wiki-use contexts, which are organizations and design projects as mentioned above. The narratives are structured in two levels: organizations and projects. For each organization, I describe the organization overview, its wiki adoption, and design projects situated in the organization. For each design project, I describe the project overview, the project team, and information sharing that uses wiki technology.

5.1.1 State University

Organization overview: The State University is one of the oldest state-supported institutions of higher education on the Pacific coast. It is a multi-campus university with campuses in three cities (and) approximately 50,000 students and 30,000 employees, (who are) organized into approximately 150 academic units, over 200 specialized research centers, and hundreds of administrative units. One of the administrative units is EdApp, an online technologies group, where two participating software design projects (Project A and Project B) were situated. This group supports members of the State University community as they discover, work toward, and achieve their learning, teaching, and research goals. At the time of study, EdApp had 15 professional staff (a director, 2 project managers, 2 researchers, 4 user experience designers, 4 software developers, a quality assurance engineer, and a marketing and outreach personnel), 3 graduate students working part-time on research, design, and technical writing, and 10 hourly undergraduate students working on testing and customer support.

In collaboration with their clients and other units on campus, EdApp explores and offers innovative technologies to help its clients stay at the forefront of their work. Figure 5-1, the State University organization chart, shows the relation of EdApp to other organizational units within the university.

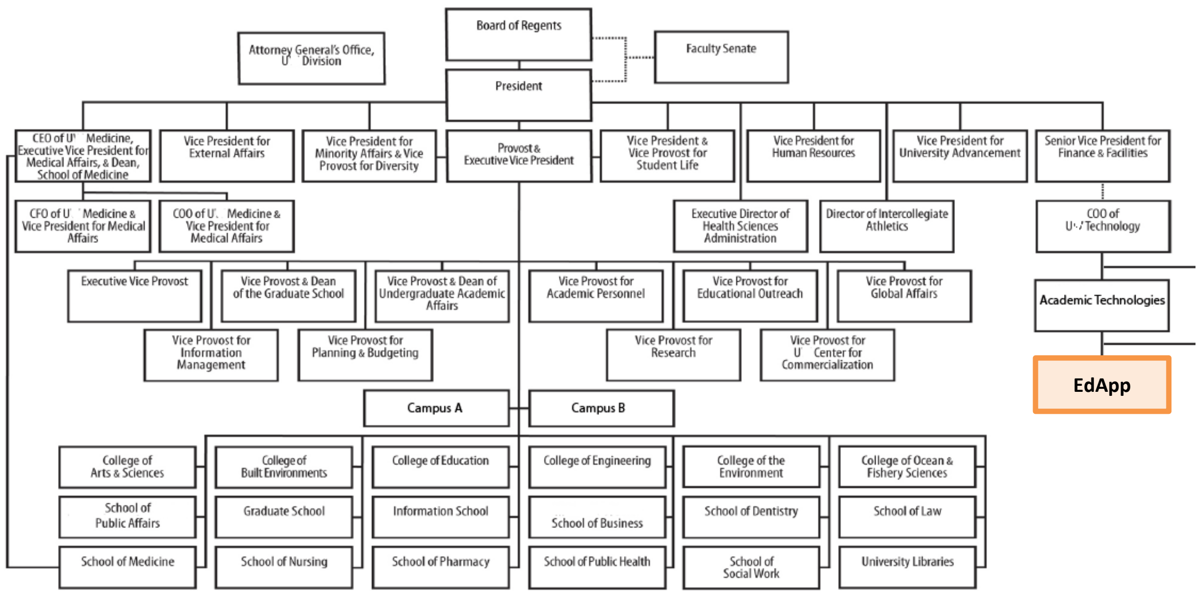


Figure 5-1: State University Organization Chart

EdApp followed a user-centered design and development process (in order) to develop web-based communication and collaboration applications for the university community. Additionally, EdApp was transitioning from a waterfall approach[2] to an agile approach for its software development process. The lifecycle for software development projects at EdApp was as follows: the planning and discovery phase (user research and functional requirements), the design phase (design, prototype, and usability study), the development phase (development and testing), and the post-release activity phase (training and workshops). Their design decisions were based on direct user feedback, user research, and findings from usability studies. The applications developed by this group had a widespread adoption by over 60,000 users from

[2] The waterfall model, originated in the manufacturing and construction industries, is a sequential design process, often used in software development processes.
[Source: http://en.wikipedia.org/wiki/Waterfall_model]

within and outside of the university. Software design project stakeholders often included users (the State University students, faculty, researchers, and staff), the university executives, other internal organizational units, and external organizations whose employees or clients use web applications designed by the EdApp group.

Organizational wiki adoption: Wiki technology was introduced to the EdApp group by its software developers in 2005, about two years before (this) study began. Prior to the use of the wiki, the group used an HTML intranet to record project information such as the project description and functional requirements. Software developers installed ModWiki and set up a developer wiki, which later caught the attention of the rest of the group. The group looked at several wiki platforms including PMWiki, ModWiki, and Confluence, and decided on MediaWiki. MediaWiki was chosen for several reasons. First, it was easy to set up and use. Second, it had features that the group was looking for, which were an ability to easily add attachments, an easy-to-view history log, and an ability to create templates to use on multiple pages.

In June 2005, the wiki intranet was set up using MediaWiki, and software development project teams started creating project wiki sites. Each wiki page consisted of: (1) the EdApp logo on the top left; (2) the sidebar navigation to functional team wikis, tools, search, and the MediaWiki toolbox; (3) page tabs – article, discussion, edit, history, move, and watch; (4) user links – username, my talk, preferences, my watchlist, my contributions, and logout; and (5) the content area. The content area changed from page to page while the other page elements stayed the same. Figure 5-2 shows the EdApp intranet wiki homepage, which was organized into four areas: Current Events (mainly meeting schedules and individual out-of-office schedules), Workshop

Events, Group Information (random items), and Projects (a list of ongoing projects). Based on the wiki log data, the EdApp intranet homepage was the most viewed page of all wiki pages. Some participants reported that they used it mostly to navigate to specific project wiki sites and to check the out-of-office schedule.

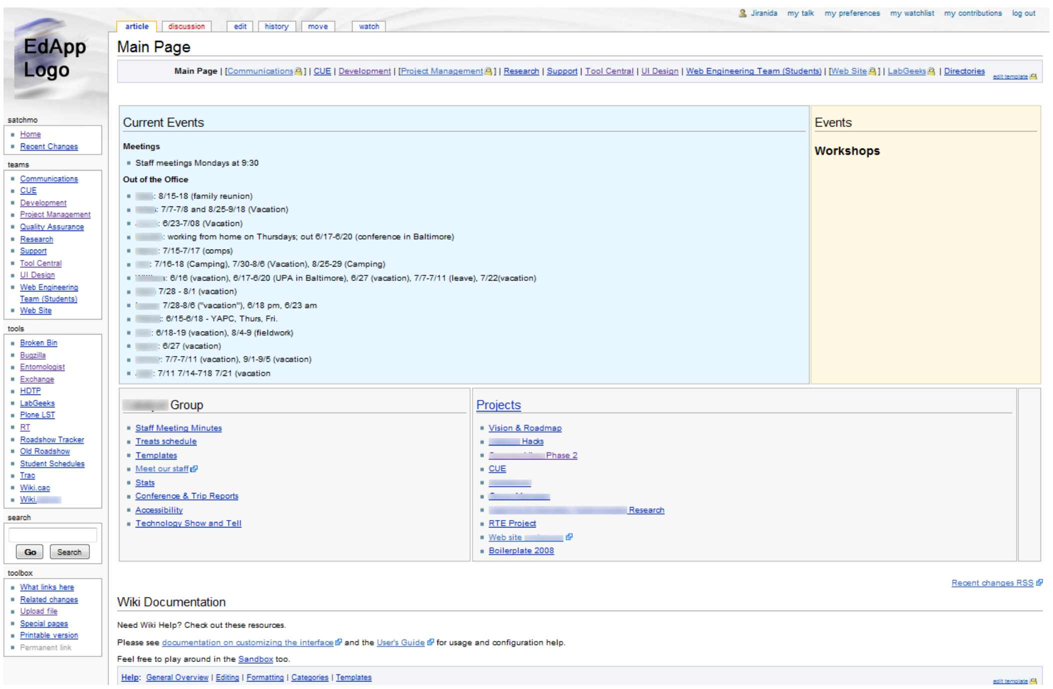


Figure 5-2: EdApp Wiki Intranet

5.1.1.1 Project A: Web-based workspace project

Project overview: The goal of this project was to design and develop a web-based collaborative workspace application that could be used for classes or projects. The work in this project was constrained largely by a project schedule and stakeholders. The project was scheduled to have its first release by August 2008, and this had a lot of influence on what and how the team performed tasks. As a project manager reported, *"One of the roles that I had was to make sure* *we were disciplined about cutting things back because we wanted to deliver something as quickly as possible that we felt good about."*

The project effort involved many stakeholders. The most influential stakeholders were the Technology Advisory Committee, the Technology Working Group (representing the outreach program, the university libraries, and several colleges), and external collaborators from several organizational units. The project effort was initiated by the request of the technology advisory committee (in order) to develop a web application with baseline courseware functionality. The working group helped the EdApp team prioritize user needs, which were identified previously from several sources.

The project team aimed to accomplish the project goal by (1) designing and developing a usable application, (2) delivering the application on time, (3) following the agile approach[3], and (4) working successfully as a team. These priorities were stated in the project definition document, and mentioned by the team members during the interviews. The team adopted a waterfall model in their software design and development process. Because the user-research work and initial functional requirements were already completed before the project team was officially formed, the project started in the design phase, where they worked on the conceptual design. General functions of the team included planning, design, development, testing, and release. The main activities carried out by the project team were creating design artifacts, design review, testing

[3] Agile software development is a group of software development methods based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams.
[Source: http://en.wikipedia.org/wiki/Agile_software_development]

design, developing code, testing the codes, and writing user documentation, and having project meetings.

To support project activities, the project team employed various tools (in order) to perform their work on this project. The team used IRC (http://www.mirc.com/), an instant chat client, to communicate with each other during working hours. At the beginning, the team used BackpackIt (http://backpackit.com/) to collaborate on the user research efforts and ShareSpaces (www.stateu.edu/edapp/web_tools/sharespaces) to store research data. Post-it notes were used mainly in a work breakdown activity, whose result was used to create a project schedule. Whiteboards were used extensively in project meetings during the design phase, in which the team met regularly to brainstorm and discuss design. The project manager used the Oracle calendar system and email to schedule meetings. Some tools were discipline specific. Examples include: project management software used by a project manager, Request Tracker (http://bestpractical.com/rt/), Entomologist (an in-house tool) and Bugzilla (www.bugzilla.org), used primarily by the quality assurance team, graphic software (e.g., Adobe Photoshop, Adobe Illustrator, and Microsoft Visio), used primarily by the designers, and SVN (a revision control system), used primarily by the developers. Other tools are office software (e.g., word processor, spreadsheet, document viewers) and a shared file server***Project team:*** Project Team A consisted of 9 members: a project manager, two researchers, three interaction designers (two full-time and one part-time), two software developers, and a QA engineer[4]. The project members' ages ranged between 18 to 45 years old, with a majority being male and under 35 years. The average

[4] Role and responsibilities of team members will be discussed later in section 5.2.3 as part of the social organization analysis.

working experience with the organization was around 4 years. All project members, except the project manager, had worked at the EdApp for at least 2 years, and had worked with each other in prior projects. A majority of project team members reported that they were comfortable using computers and wikis in their work.

Project Team A had a lot of influence and interaction with stakeholders including the committee from the university, the working group, and departments. The director of the EdApp group served on the project as a contact person between the team and other stakeholders. The director occasionally attended a committee meeting to provide an update on progress as well as bring back any concerns from stakeholders. At the beginning of the project, research scientists and designers worked together on gathering requirements. Once the project team identified user needs and finalized the functional requirements, all project members, except the QA engineer, met regularly to work on the conceptual design.

During the design and development phases, designers and developers worked closely together. They challenged each other's ideas with their discipline-specific knowledge; developers would point out technical problems with the proposed design while designers would bring up potential usability issues of the proposed technical solution.

During the testing phase, the QA engineer collaborated with software developers and designers on bug fixing. Once testing started, bug reports were shared mainly through Bugzilla, a bug tracking system. The project members, particularly the QA team, development team, and design team, shared a lot of information through bug reports after the testing period. The flow of bug reports always started with someone filing a bug report in Bugzilla. Once a bug report was filed, the lead QA engineer looked at it and assigned it to appropriate teams by flagging them with "ui_consideration", "ui_fix", "dev_fix", or "text_edit."

Information sharing on the wiki: At the beginning of the project, the project manager created a project folder in the file-sharing server and a project wiki page on the EdApp intranet wiki. The project started off with research efforts. The research team, which consisted of members from different disciplines (researchers, designers, and a project manager), created a project space in BackpackIt (an online project management tool available for free at http://backpackit.com/) to store and share research information as well as to coordinate research activities. They created a research to-do list, which a researcher said they used "*to keep track of who's doing what and who's responsible for what in a short amount of time.*" The research team also created an online file sharing space in ShareSpaces, an in-house online file-sharing application. After the user research was completed, the research team shared the user research findings with the rest of the project team via email and the project wiki. The research findings were used to develop functional requirements. The sharing of functional requirements started after the project manager had drafted the document and posted it on the project wiki page, saying *"It's the document that I'm responsible for providing on the wiki."* The document was shared in order to gather feedback from different groups. The team reviewed the functional requirements in a team meeting and finalized the requirements together. The project manager revised the functional requirements document four times and uploaded the updated versions to the project wiki. The final version of the functional requirements document was then used by designers to do their design work.

The project manager then created and shared the project schedule on the project wiki page. He also sent an email with a link to the schedule wiki page to the team, notifying them that the schedule had been posted. Designers started working on the design specifications by creating wireframes and screen mockups. Wireframes and mockups were shared in the file-sharing server. It was an iterative process in that the designers shared wireframes and mockups with the team to get feedback and made changes based on the feedback. When the mockups were ready for review, the project team reviewed them in a meeting. After the team agreed on screen mockups, the designers created the design specifications document on the wiki. Occasionally, in a project meeting, the designers gave a status update on the design specifications to the project team.

The QA lead read through the design specifications on the project wiki to help himself develop test plans. The test plans were step-by-step instructions for the tests that testers should complete on a particular application. Interestingly, test plans were never posted on or linked from the project wiki even though there was a link (placeholder) for them under the QA section. The QA lead reported in the interview that he did not feel like test plans needed to be shared on project wiki sites.

During a release day, the QA team and the development team coordinated via the next push checklist. The next push checklist was a wiki document that contained a list of what developers had to do for the next release as the QA lead described, *"[It] is a single page of, for our next release, what exactly do we need to do because all of our releases are more than just copying code onto the server. Most of the time, there's some database change that needs to happen, there're configuration changes that we'll need to make on the server on that day."* All the changes that had to be made to the production database and codebase (like the exact SQL statements) were logged on this wiki page. The QA lead used this next push checklist to simulate the entire release prior to the release date. It made a release day smoother because the developers and the QA engineers could easily make sure everything actually got done for each release day. Completed items were marked off as they were working on it during the release. As a result, the down time and the number of errors during the release process had been reduced. The next push checklist was not for any specific application or project; it was for cross-project changes that were to be released at the same time.

5.1.1.2 Project B: Online discussion board enhancement project

Project overview: This project was a multi-release project with three releases. Its goal was to add more innovative, advanced, and administrative features to an online discussion board, which was built with basic features and released approximately a year prior to this project. Thus, the work in this project was constrained by pre-defined user requirements and technical constraints from the existing application. The user requirements were carried from the first version of the online discussion board.

The project team aimed to accomplish the project goal by (1) incorporating user feedback on the current online discussion board application, (2) releasing additional features on schedule in multiple releases, and (3) working successfully as a team. Team B did not follow a full project life-cycle for its design development process because this project was a multiple-quick-releases project. The project was divided into three overlapping phases. Each phase consisted of the following activities: design, development (back-end development), template work (front-end UI development), testing, and technical writing. General functions included planning, requirements gathering, design, development, testing, and release. The main activities carried out by Team B-were prioritizing functional requirements, planning a project schedule, creating a design specification, conducting a usability study, developing codes, testing codes, and planning releases.

Team B used a variety of tools to accomplish their work. The team members reported during the interviews that they did not use email much because all team members were located in close proximity, and that they preferred a face-to-face interaction. One developer reported that he got *"very few emails,"* which were mostly about scheduling a meeting. Another developer on the same team said that he did not use email at all in this project. A designer on the project also reported that she did not get any email from the team.

The designers and developers used IRC, GoPost (an online discussion board), and Gobby (a collaborative authoring tool) to communicate during the design and development phases. One of the designers archived his work files in a project folder on Satchmo, a shared file server. WebQ (an online survey tool) was used to develop an online survey to collect user data to help the team prioritize some functional requirements in the last release. Team B gathered requirements from bug reports in Bugzilla, and from user requests in Request Tracker (RT System)[5]. Bugzilla was also used during the testing phase to report and fix bugs.

Project team: Project Team B consisted of 6 members – a project manager, two interaction designers, two software developers, and a QA engineer. The QA engineer also worked on

[5] Request Tracker (RT) is a ticket-tracking system that the EdApp group was using to store and manage help desk issues, which included user feedback, new feature requests, and technical problems.

Project A. The age of project team members ranged from 18 to 45 years old, with a majority of members being under 30 years old. In addition, their work experience at EdApp was lower compared to those in the Project Team A. Three project members had worked at the EdApp for approximately 4 years while the other three had worked there for less than 2 years. However, the project team was high in both technical skills and wiki experience, with only one designer having little prior wiki experience.

The project team did not have researchers because the project scope did not require that user research be conducted. Instead, the team gathered requirements from user feedback and bugs that already existed in the RT system and the bug tracking system. The designers were responsible for both research and design tasks. One of the designers took the lead on drafting the functional requirements documentation, which was reviewed by the project team in a series of project meetings. The project team also worked with an undergraduate student who was responsible for writing the help documentation. The designers and developers worked closely together during the development phase. They collaborated through Gobby, a free collaborative editor that supports multiple documents in one session, and a multi-user chat application. It runs on multiple platforms including Windows, Mac OS, Linux, and Unix-like platforms. With Gobby, there is a window that all users share, can type into it, and see what other people are typing. A to-do list was created to keep track of things that needed to be done during the template work. They also used a chat feature in Gobby as a communication channel to inform each other about what needed to be changed either in the design specifications or in the codes.

Information sharing on the wiki: One of the designers drafted the functional requirements document on a wiki page, which contained associated bug report numbers and RT numbers (if applicable), a description, and time estimate. The project team reviewed and discussed the requirements in three project meetings. They pulled up the requirements wiki page on a big display. A project manager created a project schedule based on the time estimates on the requirements wiki page. The schedule file was created using a project management program, and kept in the Satchmo file sharing server. The schedule was updated with new dates as the project went on. The project manager converted the schedule file to an html file and posted it on the project wiki to make it look nice on the wiki page. The project team used the project schedule to keep track and update work progress, which had made the development tasks completed faster than the estimates as she explained, *"But once I posted the html schedule up there, they actually used it to check things off once they're done. For one project, the developers went through development tasks five times faster than what they had estimated. So, it's very motivational to check off this is done, this is done, done."*

The designers in Project B started creating and sharing a design specification much earlier in the process than that of Project A, as one of the designers explained, *"Because before we were working on stuff, and we didn't want to put in the spec until it was done. That's usually how we do the work. But now I feel like it's much easier just to make the spec as a working document so you can start organizing it. Then there is one place to go. We don't have to transfer stuff to a new spec page. And then just to note the status of it at the top."* So, the designers created a wiki page titled "[Project name] Design Specification" on the project wiki where he started composing the design specification for phase 1. The designers also decided to make much bigger thumbnails for screen shots, and place them on the left side with explanatory text on the right. In addition to the specification content, the designers also put status information on top of every page to indicate progress for each feature. The status information says, for example, "In progress," "Done [date]", and "templates done [date]." While the specifications were still in progress, the designers asked the developers to review and give feedback. They met to discuss the issues the developers had, and if necessary the designers would revise the specifications to reflect the developers' feedback.

They had twenty-five meetings recorded on the meeting notes wiki page. These included both formal project meetings and informal meetings. Meeting notes were usually taken live on the wiki during a meeting. Someone would pull up a meeting notes wiki page on a big display and typed as a meeting progressed.

5.1.2 Flynow

Organization overview: Flynow is an air carrier that serves more than 16 million passengers annually to over 90 destinations in the United States, Canada, and Mexico. Founded in 1932, the company has grown and now has over 10,000 employees worldwide. With its spirit of technical innovation, the company often has ongoing IT system design and development projects. The organization's Information and Technology department reports directly to the company president. Under the Information and Technology department, the IT Applications group provides computer applications and business intelligence tools to Flynow. Although most software development projects are managed by the IT Applications group, they often consist of people from different organizational units such as Information and Technology, Marketing, and Customer Services. The Flynow organizational chart is presented in Figure 5-3.

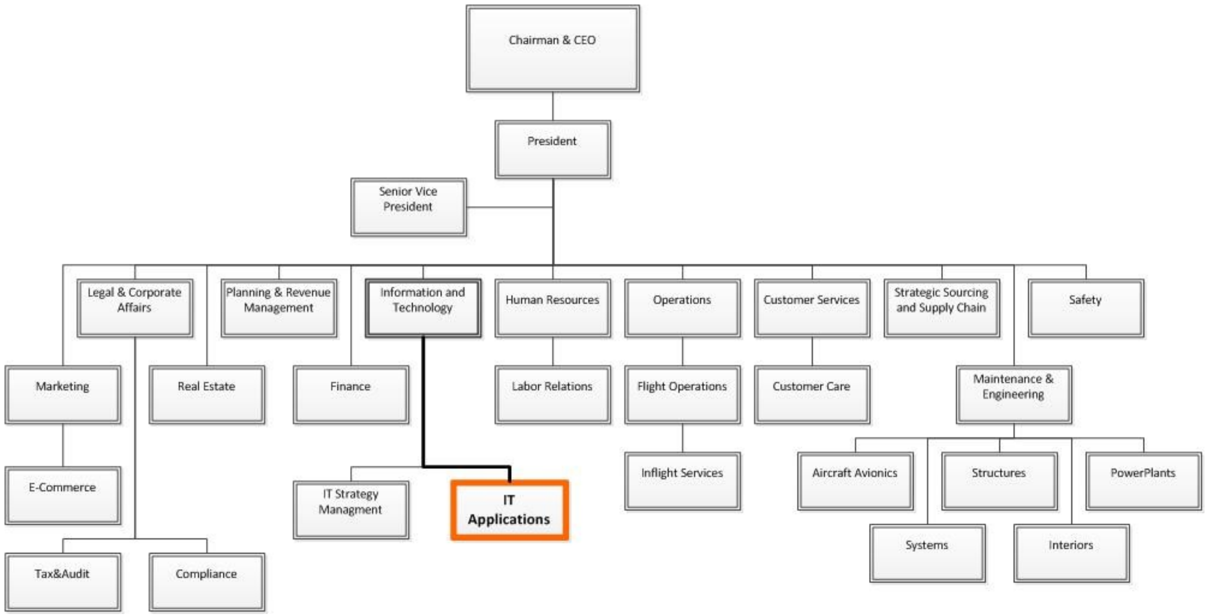


Figure 5-3: Flynow organizational chart

Flynow has offices around the world, but all software design and development projects are done at the company headquarters, which consisted of several buildings within a close proximity. At the time of study, the majority of organizational units were collocated within the same building, with several units having recently moved to new buildings located several miles away.

Software design project stakeholders often included the project members, which came from various organizational units, executives, outside contractors, and product vendors. There were also Federal Aviation Administration (FAA) regulations and policies that needed to be considered when carrying out any IT projects.

Organizational wiki adoption: The wiki adoption at Flynow started within a software development group when developers installed and used a wiki to share information internally.

The use of wiki gradually spread out to other groups in the organization. The organization officially adopted wiki in early 2006. The wiki was centrally managed and supported by the Information Technology department. Since then wikis have been used successfully in many software development projects.

Flynow used Openwiki (www.openwiki.com), a text-based wiki, because it was already used by early wiki adopters. The organization had an index site of all known wikis at Flynow, which was called "The Mother of All Wikis" as shown in Figure 5-4. There were approximately 150 wiki spaces at the time of study. The process to create a wiki space was sending an email to the IT department with a suggested wiki name, as described by a project manager, *"We just have a process where you request a wiki from our IT department, and they just basically set up this front page in blank, and it's up to you to organize the information."* The IT department did not allow file uploads into the wiki space in order to prevent a data storage issue. Thus, wiki users needed to store files elsewhere and to link to them from the wiki.

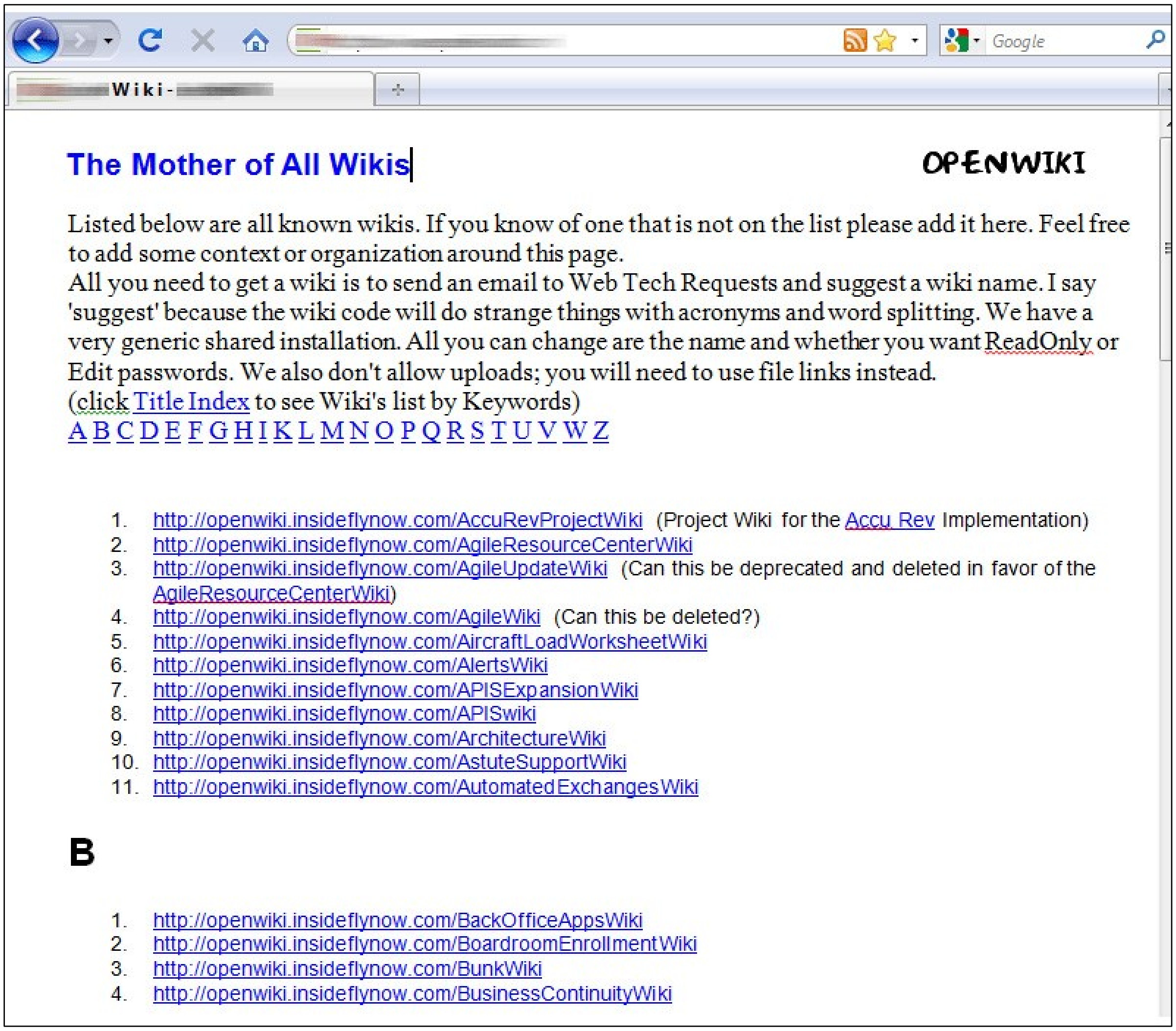


Figure 5-4: Flynow's Intranet Wiki site

5.1.2.1 Project C: Ticket Reprice and Reissue System (TRRS)

Project overview: The goal of the TRRS project was to develop a web-based application to support self-service ticket reissuing in order to save cost and reduce the workload of the company's reservation office. The project was sponsored by an e-commerce group within the marketing department. It was budgeted to be a 10,000-hour project with eleven team members from multiple departments. Major constraints for Project Team C were the company website, stakeholders, team structure, and the agile process. The project consisted of four phases before the release. Team D followed the Scrum approach[6] (an agile software development method) as they worked on this project.

The project team had set priorities to (1) design and develop software that follows the business rules and is usable for customers, (2) deliver the software on schedule within an estimated budget, (3) collaborate and communicate effectively with stakeholders, and (4) follow an agile development process. Project functions included managing, design, development, testing, release, and communication. The TRRS project was development and test driven as reflected in their team structure; 3 developers, 4 QA systems analysts, and 1 interaction designer. Thus, a lot of effort was put into testing. Major project activities were creating the project charter, managing tasks, creating business rules, designing interaction and user interface of the software, coding, testing, communicating to executives and internal stakeholder, coordinating with vendors, and releasing the code to the company website.

The project team heavily used a spreadsheet application to create project documents such as business rules, test scenarios, test scripts, and bug lists. The project manager created a project burn-down chart, a graphical representation of work left to do overtime in order to track the project work and efforts, and called it "Task Manager." The project team used the wiki to create a project portal page where they linked to documents and folders in the file-sharing server. Other tools that the project team used include FirstClass (an email client application), Mantis (a

---

[6] Scrum is an iterative and incremental agile software development method for managing software projects and product or application development.
[source: http://en.wikipedia.org/wiki/Scrum_(development)]

bug tracking system), graphic design applications, and office applications (Word, Excel, and PowerPoint).

Project team: Project Team C consisted of 11 members: a project manager, a product manager, three QA system analysts, 4 software developers, an interaction designer, and a pricing analyst. There were 4 males and 7 females, with a majority over 36 years old. Most of the project team members had worked at the Flynow organization for a long period of time (the shortest period of 4 years and the longest period of 22 years). Although most of the project team members considered themselves highly technical, they reported that had little to no wiki experience prior to working on this project.

The project manager was responsible for managing the team and communicating with a steering committee that helped make high-level decisions. A product manager, who was also a project sponsor, played a crucial role in developing business rules and use cases. There was one interaction designer while the majority of the team members were software developers and QA system analysts. The project manager explained that this project had more QA systems analysts than normal software projects at Flynow and that they spent more effort on QA than the actual development. When the project scope was expanded, the team added a price analyst from a refunds finance team to the project to provide insight on the re-pricing process. The project manager worked with representatives from each disciplinary group on finalizing the business rules.

The project team also collaborated closely with a vendor, a provider of innovative solutions for the travel industry on the East Coast. They had a list of bugs and a resolution to communicate with the vendor. The vendor emailed the project team a weekly spreadsheet, in which the team prioritized a feature list for the vendor.

Information sharing on the wiki: Once the project started, the project manager created a project folder in the file-sharing server and requested a project wiki space. Documents in a file-sharing server were organized by projects; each project had a folder. The project wiki site then became a portal for accessing a project folder. The project manager accessed the project folder more often than the rest of the team (at least a couple of times a day because she was *"the keeper of most of the documents there"*). Confidential information would not be shared in the file-sharing server. For large documents, project members would share them in the project folder in the file-sharing server, and created a link to the file on the project wiki. Sometimes the sharer would notify other project members of the shared file via email or in a Scrum meeting. The project team used email extensively. The project manager estimated the amount of at least 1,000 emails sent among the project members from January to August 2007. Email was mainly used to set up meetings and for discussions.

The project wiki served as a portal for accessing project information. It had links to many documents (e.g., use cases, mock ups, task manager tool, bug lists, business rule documentation, the vendor's list of issues) and several information-sharing tools (the project folder in a file-sharing server and a bug tracking tool). The project wiki was accessible to all project members and to the entire organization through the "Mother of All Wikis" intranet page, but it was not accessible to the vendor. The project team, however, had access to the vendor's bug tracking system.

The project manager requested information about tasks and estimates from each sub team based on their expertise. The project manager planned the development cycles, and created a project task manager and schedule. The team called a spreadsheet for task management, “a task manager tool” as labeled on the project wiki homepage and mentioned in interviews. The project members were collocated on different floors in the same building. With the agile software development cycle, communication was mostly through face-to-face conversation. The project team met daily in a 20-minute meeting, the Scrum meeting, during which the team would review the work breakdown in the task manager and went through their daily tasks.

At the beginning of the TRRS Project, Team C was working on understanding business rules. The business analyst team documented business rules in a spreadsheet, and gave the documentation to the product manager. The product manager put the spreadsheet file in the file-sharing server as well as created a link to the file on the project wiki homepage. During several project meetings, the whole team reviewed and discussed the business rules together using a big screen as they developed use cases on the project wiki. Because it took too much time formatting the use cases on the wiki, the team ended up composing it in Word, and linked the document to the wiki. The use cases were revised many times until the whole team had agreed. Then representatives from various groups got together to go through use cases and planned the development cycles. This happened in a series of meetings where they hashed out what needed to happen.

The interaction designer would access the use cases via the project wiki when designing interaction flows and user interfaces. Software developers and QA systems analysts used the use cases to plan their development and testing tasks. The team members would refer to the business rules document to help clarify any issues in the use cases. The project team stopped referring to the use cases once they had captured all the requirements, which included issues, logs, and tasks in other documents as the project manager described, *"But then once we get to a certain point in the project, we kind of drop the use cases because we've captured what we need, and then we basically have issues, logs and tasks that sort of take over the use cases."*

The interaction designer started the design task by going through the existing system to identify which parts would change, and also to design for consistency. She dealt with the user interface side of the project. For example, on the flight availability page that needed to show prices, she created a mock up for a new design and developed an interaction flow. The sketches were done in Photoshop and PowerPoint. In order to do her design work, the interaction designer needed to know the business rules and technical information to understand the restrictions. The output of design work was screen mockups and a blueprint (design specifications), which were shared on the project wiki. The blueprint described how the screens should behave. It contained different use cases, design rationales, and descriptions of the interaction. The interaction designer also conducted a usability test with customers, thus she shared usability testing information with the project team in case anyone was interested. The interaction designer explained that she shared design information to other project members so that *"[the developers] know what they needed to code, and also to make sure that [she] was following the business rules."*

Developers primarily used use cases, screen mockups, and the blueprint to develop the code. The developers would ask the interaction designer for clarification if they did not understand the design, either via email or face-to-face. The interaction designer said that she sometimes *"watched over their shoulders"* when they were coding to ensure that they were developing the right code.

With help from Project manager, the QA team created test scenarios in a spreadsheet file, which showed test cases, estimated testing hours, and actual testing hours. The information was also presented visually in a line graph, which was called a burndown chart. The document helped the QA team track their work and communicate their progress to the project manager. Issues found from testing were logged as bug reports in Mantis[7], a bug tracking system. The QA team was responsible for assigning bug reports to developers or the designer. Once the bugs were fixed, they would be assigned back to the QA for verifying.

5.1.2.2 Project D: Virtual Agent (VA)

Project overview: The Virtual Agent project's goal was to develop an online, voice-enabled Virtual Agent to help customers find information on the company website, which was supposed to lower calls going to the call center. The Virtual Agent's job was to answer customer's text queries both verbally and in writing. In addition, the software could ask customers follow-up questions in order to provide useful answers. The project was part of the company's spirit of innovation, which led it to be the first airline in North America to offer such services. The software was launched to the public in February 2008. Within six months after the launch, the Virtual Agent was already handling roughly 3 percent of all interactions between the company and its customers. By February 2009, a year after the launch, the program had handled more than

[7] Mantis is a free web-based bug-tracking system. http://www.mantisbt.org/

4 million customer queries. As a result the company decided to eliminate customer care Web chat on its website for cost efficiency.

Similar to Project C, the first priority of Project D was developing and delivering the Virtual Agent software to the company website on time. The Project Team D had two major constraints. The first constraint was the project scope because, being an innovation project, its scope was not as well defined as that of a normal software development project. The second constraint was external collaboration because any changes or updates in other organizational units or to the website could affect the designed software. Therefore, working within such constraints, Project D's priorities included (1) getting buy-in from the whole company as explained by the project manager, *"This is a little bit different because it's an innovation project. While there was still a timeline and tasks, my bigger role was defining what it actually was and selling it to the organization,"* (2) collaborating effectively with internal stakeholders (other organizational units) as the product manager stated, *"One of our big goals is to make sure that we are staying in contact with all the product and project managers so as they're bringing out new information, they're bringing out new products to our customers onto the website that we're synching up with them in the beginning of the projects so that we can let them know how we can help,"* and (3) collaborating effectively with external stakeholders, particularly an international software vendor.

The team did not fully adopt an agile software development process, but followed an iterative approach during its development and testing phases. The project was 18 months long, started with scoping and planning back in September 2006, and finished with a public release in February 2008. Project activities were not focused as much on designing software interaction and user interface like a typical software design and development project, but rather on defining the project scope, designing and creating a knowledge base for the Virtual Agent, collaborating with a third-party vendor, integrating the software into the company's website, and testing the software. Therefore, project activities included, for example, conducting a focus group, scheduling project meetings, creating the virtual agent's dialogue, recording the virtual agent's voice, the installing development and QA environments, testing the software in each environment, and releasing the code to the company's website. Similar to Team C, Team D used various tools including a wiki to create a project website, Mantis (a bug tracking system), a file-sharing server, and FirstClass (an email client).

Project team: Project Team D consisted of a project manager and 13 project members grouped into four sub-teams: E-commerce, Employee-facing development, Customer-facing development, and IT services. These project members belonged to multiple departments including IT strategic management, Marketing, IT development, and Customer care. Most of the project team members were female (12 female and 2 male), with at least half 36 or over. The majority of project members had a long working history at Flynow, ranging from 4 months to 20 years with the average of 7 years. As reported in interviews, the level of technical expertise and wiki experience varied among the project members. Five project members reported that they did not use the wiki technology prior to starting the project.

Since the project was an innovation project, the project manager's additional role was to define the project and to sell it to the organization. The e-commerce team was responsible for communication between the marketing business group and the project. One of the e-commerce team members also served as a product manager on this project, whose role was to oversee that the project matched well with customer needs. There were two development teams on this project: the customer-facing development team who worked primarily on developing the knowledge base of the virtual agent as the team lead called it *"the brain"* of the product and the employee-facing development team who worked on implementing the virtual agent software. The eCommerce team worked with the business side to document the business rules. The business rules were discussed between the project manager and the vendor who developed a virtual agent solution package.

Information sharing on the wiki: When the Virtual Assistant project started, the product manager who was from the eCommerce team requested the project wiki site. Once the project wiki site was created, the project manager created the content and organized the project wiki page. She structured the project wiki homepage using the structure of the project notebook, which was a required document for all projects by the Project Management and Organization department. It included the project schedule, plan, the status report, role and responsibility document, and other project management details, most of which were linked from a project folder in the file-sharing server.

The project manager explained that specific key documents that were critical and often changed a lot would be linked directly to the documents themselves. For some documents that were part of a series like weekly status reports, she created a link to a folder instead of to specific documents. She also had a link to the project folder on the project wiki. The project manager described that linking to documents on the project wiki was a way to broadcast information to the entire project team. For example, the project manager scheduled a meeting and published a calendar of meetings for the week on the wiki, expecting the project members to review the calendar as she said, *"It's their responsibility if they think they should be included, at least it's broadcasted."* However, when she would like the team to review certain documents, she would send an email to the team with a link to documents. The product manager did not link unfinished documents on the project wiki site as she explained, *"I typically have a personal folder where I keep things that I'm developing at the time, so I don't put half-finished documents on the wiki. I wait until they're all finished, and then I move them over, and then you don't have six versions of something."*

The eCommerce team had never worked with the project wiki before. They had relied on a file-sharing server *"with lots of folders to keep projects in, keep it in this folder, keep it in that folder,"* which they found as a great way of looking at things from a company perspective. The eCommerce team used Mantis to keep track of work progress, to keep people updated without using email. They also used Mantis as a discussion board. A project member from the eCommerce team reported that the eCommerce team did not access the project wiki site very often; they primarily used the project wiki site as a bookmarking tool for accessing links that the team frequently used.

Information on the project wiki used by the customer-facing development team included project definition, mockups, business rules, and project planning. The customer-facing development team accessed the project wiki site several times a week. They explained that they accessed the project wiki site via a shortcut that went to *"the folders that [we] created and that [we] used a lot"* because it was easier than to *"actually go to the front page of the wiki and finding the information."* The customer-facing development team was a contact point between the project team and other organizational units as they requested information from other units that would affect Project D as well as informed others about the project release timeline. They also exchanged information with the vendor because the vendor was coding required questions and answers that the team had created.

5.1.3 Digital Media Company

Organization overview: Digital Media is a leading creator and distributor of images, footage, music, and other digital media content, providing the world's largest imagery collections. It is a global company with approximately 2,000 employees and twenty offices worldwide. Digital Media has followed an agile process for their software development with a 6-to-8 week release cycle. Most software development projects were managed by the Application Development (AD) unit, which was under the Technology department, as shown in the Digital Media's organization chart (Figure 5-5). The AD unit was responsible for building the company's primary consumer sites and revenue generating products including search infrastructure, UI, commerce, and web services. These sites were 24x7, high volume, zero downtime, e-commerce sites supporting 7 million searches per day. The AD unit was organized into various teams by functions or products and services such as Search, User Interface, Shared Technology, Performance, Architecture, Integration, Quality Assurance, Legacy, User Experience, and Enterprise Web Services.

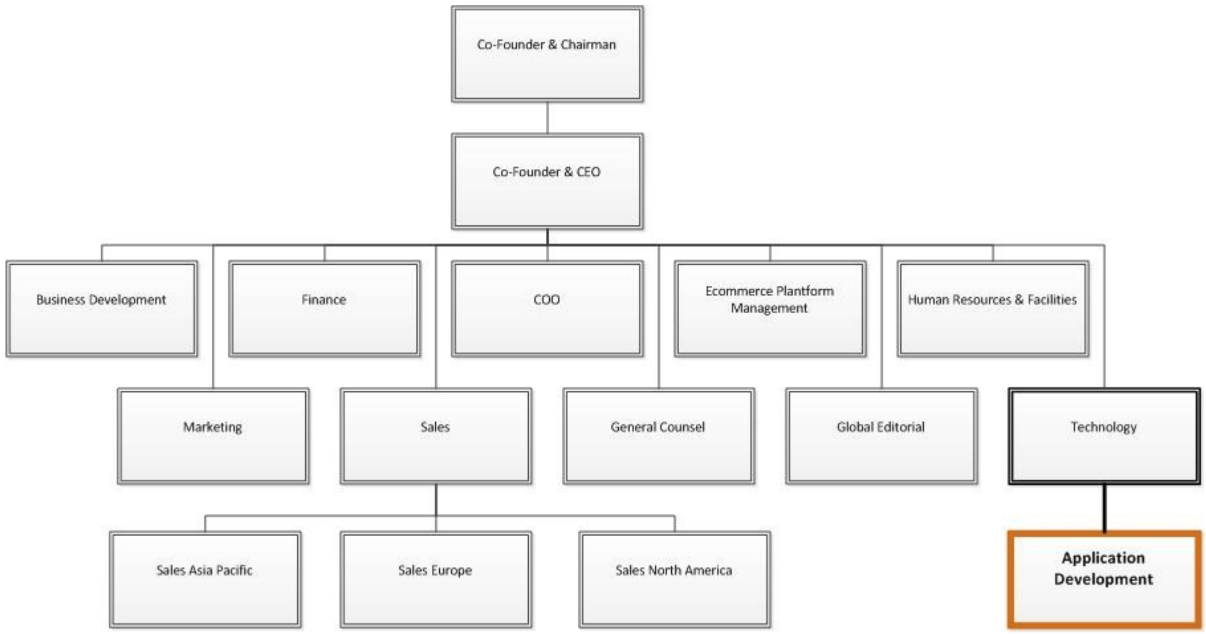


Figure 5-5: Digital Media organization chart

Software design and development projects were managed through the technology unit; however, each project consisted of members from different organizational units. Software engineers and quality assurance engineers (testers) were located on the same floor while other units were on different floors of the same building. Their projects often involved stakeholders from outside of the company, which were editors, photographers, contractors, and product vendors.

Organizational wiki adoption: Digital Media's engineering group adopted wikis for their internal use in 2005. Prior to wikis, the organization had its intranet based on SharePoint[8].

---

[8] SharePoint is a Web application platform developed by Microsoft (http://sharepoint.microsoft.com/en-us/Pages/default.aspx). One of its most common uses is an Intranet portal to centralize access to enterprise information and applications on a corporate network.

Interview participants at Digital Media reported that people were not happy with SharePoint's functionalities and capabilities, so some organizational units created their own wiki as explained by a QA lead engineer, *"Before we had SharePoint, and trying to find anything on SharePoint is a real pain. They are all in individual documents. I think the major issues were SharePoint isn't friendly when it comes to capturing information and posting. Second of all, some people just took their own initiative to create their own wiki. [A developer] created [a wiki site] on his own box. [A sales engineer] created the blogit one, and I don't know who in the [Network operations center] created their own."* Thus, at the time of study, several different wiki platforms were deployed by different functional groups at Digital Media. Those wikis were Development Wiki (MediaWiki) used by developers and QA engineers, PMO wiki (MediaWiki) used by project managers, NOC wiki (Blogit) used by network engineers, TS wiki (MonMonWiki) used by technical services engineers, and Sales wiki (MediaWiki) used by customer support and sales representatives. There was an effort to reorganize and integrate different wikis into one central wiki to use company-wide.

5.1.3.1 Project E: Customer Web Experience

Project overview: The project's goal was to improve the customer experience of the company website. In summer 2007, the team worked on integrating a music agent service, provided by another company, into the company website so that corporate media advertising companies can search for and purchase music online. However, due to an executive's decision, the team was moved on to a different project, called customer facing web zone. The project consisted of three parts: (1) supporting the development efforts of the Web photography flash application; (2) developing chat on the search results page; and (3) implementing a Google sitemap index.

Major constraints for the team were organizational factors, including current product and services (the company website), the software development process, and intra-team collaboration. The organization had chosen to adopt a Scrum approach (an agile development process) with a 7-week release cycle and a week-long sprint, which Team E followed. Following the Scrum approach, Team E set priorities to (1) develop and release selected features to the company website on schedule and (2) communicate frequently and effectively among the team members. For example, the team started working on the Web photography flash application in mid-June 2008 with the first release to production scheduled in early August 2008. The team had to extend a release during the Olympics 2008 because the management did not want any changes to the website during that time, as one of the developers explained "*Yeah, we call it a release, and that lasts about seven weeks, so this one's a little bit longer because we have the Olympics right now, and [Digital Media] is the additional photography provider or an image provider for the Olympics, so they don't want us touching the website during the Olympics because it's very… We're high profile.*"

In addition to traditional software design and development activities, Team E had Scrum-related activities such as daily Scrum meeting, sprint planning meetings, sprint review and retrospective meetings, creating a product backlog, creating a sprint backlog, and creating a burn down chart. The team used ScrumWorks, an agile project management application, to support Scrum activities. While wiki was used as a central information-sharing tool, the team also shared information via SharePoint.

Project team: Project Team E consisted of 11 core project members: a project manager, 5 software developers, 3 QA engineers, a UI designer, and a linguistic specialist. There were more male than female members (8 males and 3 females). The majority of project members had worked at Digital Media for less than 3 years. Their educational backgrounds mostly were in the science and technology fields. All project team members were very comfortable using computers for their work; however, approximately half of them had no or little experience with wiki.

It is interesting to note that the Application Development unit, which comprises software developers and QA engineers, had been reorganized shortly before the project began. The unit had changed from grouping developers and engineers by features such as search, footage, legacy, and editorial tools, to four groups, which encompassed people from across prior feature groups. The new grouping was aligned with a self-organized-team concept in an agile approach as a software developer explained, *"So now, it's more instead of grouping people that way, you know, by feature, it's more of a top to bottom thing. You can do the front end, but you also have to do the database, the back end, the middle. It's a vertical slice versus the horizontal slice that we were doing before."*

Following a Scrum process, a project manager also served as a product owner, and one of the software developers served as the Scrum master. In addition, the project manager was also responsible for developing business requirements because the company does not have a business analyst position.

During the planning phase, the project manager, who also held a business analyst role, provided the analytical information to the business people and translated business requirements to technical requirements. The project manager was responsible for negotiating the business requirements with the business and steering committee management. The development team was responsible for generating the detail requirements (technical requirements), which would be broken out in the high-level requirements. The planning phase concluded when the project team members articulated what work needed to be done based on the high-level requirements, and the project manager used that information to reevaluate the plan if needed. Once the detail requirements were generated, the design team created some screenshots and conducted a usability test.

The project team had a daily Scrum meeting, during which team members reported to the team what they did and what they plan to do. The project team worked closely with 5 other teams from different units, which were the asset management team, the integration team, the Oracle financial team, the Alliance royalty team, and the Cataloging team. The project manager was responsible for coordinating with each team lead, and communicating back to the project team.

Information sharing on the wiki: The project manager documented high-level requirements (an architectural business roadmap) on the project wiki site. He occasionally brought up any issues that needed re-prioritization to the business stakeholders. The project manager was very flexible in how the high-level requirements were communicated as long as they were useful for the people who needed them. He explained that sometimes he would deliver the business requirements verbally during a quick 5-minute conversation while other times he would create a formal document.

The technical requirements were documented on the project wiki site to create a product backlog that the team could manage easily. For each release the project manager (also a product owner) and the Scrum master reviewed the backlog on the wiki page, evaluated the priority, and assigned the order in which the team should work on for that release. The Scrum master then took the prioritized list and put it into ScrumWorks. After that the project team reviewed the stories and tasks stored in ScrumWorks, and discussed the actual work that needed to be done.

In ScrumWorks, there were epic stories (ones that spanned multiple sprints), small stories (ones that could be accomplished within a sprint), and tasks. Small stories were not captured in the project wiki. However, for epic stories that took a larger effort, the team needed the wiki as a better tool to track their work and kept them from getting out or expanding the scope as the project manager explained, *"But for the case of a lot of the larger efforts that we undertake, we really need the wiki as a single place to go, to really refresh our memory or keep us on track, keep us from getting out or expanding the scope of what we're trying to do, etcetera. It's a nice tool to capture the larger vision, much more so than ScrumWorks ever could be, mainly because of the size of what we're trying to do."* The QA team developed test cases based on stories in the ScrumWorks. The stories had associated acceptance tests that the QA team used to guide their test cases and test scripts. The project team had a daily Scrum meeting, in which all project team members met, reported what was done the day before, and planned what was to be done the following day.

Many project documents were stored in SharePoint. The project manager converted certain documents such as architectural documents, created by the architecture team, into wiki documents. The reason that the project team used SharePoint was that the finance department required that all business-related documents be stored in SharePoint. The architecture team had a legacy process to create a formal document using a word template and to store it on SharePoint. The architectural documents were used by the finance team to justify how the project team spent hours working on the project. The finance team was *"expecting that documentation in a formal manner [was] out on a SharePoint site in a particular place."*

The project manager had gotten permission to use the wiki as an authoritative source of project information, from planning all the way to lessons learned at the end of the day. There were two wiki sites that the project team used for sharing and managing information. One was created by the Project Management and Organization (PMO) department, and the other one was created (and used) by the development and QA departments. The directors and VPs of the PMO department put together a main PMO homepage and set up a project site for every project in the company. The project wiki site was a modification of a Word document template the PMO department had used. The development and QA wiki was linked to the PMO project wiki. It was used primarily by the QA team. The QA team used the development and QA wiki internally to communicate project level status across the QA department. The QA team had a project dashboard, for all projects they were assigned to, where they track performance for each release. According to a QA engineer on Project E, the QA team preferred verbal communication; they liked to have a face-to-face conversation when sharing QA related information to other project members. When they got a request from the development team about the performance of a feature being developed, the QA team would send an email with the requested information. There were certain documents, mostly Word and Excel documents, which the QA team stored on SharePoint.

The project manager expected a PMO project wiki to be a repository for project information as he expressed, *"At the end of the project life, the whole project site should tell the entire story of the project."* He also included the initial presentations and thoughts about the project as background information on the project. The project wiki was structured to reflect the standard project phases (i.e. planning analysis, design, development, and testing), and was envisioned to evolve as the project progressed. The project manager used the project wiki (that he created) for status reporting. He updated weekly statuses with a color-coded scheme (red: not started, yellow: in progress, green: finished). He also created the updated executive summary, which would go to the business stakeholders and other executives who might be interested. As a result, the PMO project wiki site was geared toward the steering committee and the stakeholders from the business side, rather than a working space for the development and QA teams.

5.2 Context Analysis

This section describes a collective context by highlighting similarities and differences between individual research sites presented earlier in section 5.1. The collective context is described in terms of the CWA dimensions, which are the environment, work domain, social organization, activity, and actor's resource and values. Table 5-1 demonstrates how different components of individual contexts are mapped to CWA dimensions.

Table 5-1: Context in CWA dimensions of analysis

CWA dimensions	Components of individual contexts
Work environment	Organizations
Work domain	Projects
Social organization	Team's structure, division of labor, and collaboration
Activity	Team's information sharing on wiki
Actor's resource and values	Team members' experience, perception, and attitudes

5.2.1 Work environment: Three organizations

Despite the fact that three organizations were in completely different industries – higher education (State University), airline industry (Flynow), and stock photography (Digital Media) – they represent a common work environment for a typical software design project situated in a large organization with thousands of employees. These organizations are similar in several ways. First, they had an IT department as a designated unit responsible for software development projects. Second, at the time of study, they had recently adopted an agile approach for their software design and development projects. Third, their wiki adoption began as a grassroots effort.

An Information Technology department (IT department) is one of the top-level organizational units represented in the organization charts of all three organizations, which implies its importance to all of the organizations. The IT department is responsible for internal IT development (strategic oversight, planning and direction of the organization's IT infrastructure, resources and services) as well as external IT products and services (planning, managing, and developing applications and services for the organization's customers). Within the IT department, there was a subunit that was responsible for developing software applications for the organization's customers. That is, the EdApp group for State University, IT Applications group for Flynow, and Application Development group for Digital Media.

At the time of study, the IT departments of these organizations had recently adopted an agile approach for their software design and development projects. All of them were still new to the agile development approach. They previously followed a waterfall development model, a linear and sequential approach to software design and systems development, with which each stage of the project was assigned to a separate team. With the agile development approach, software design and development teams worked in shorter cycles (a week for Digital Media, two weeks for State University, and a month for Flynow). The project priorities were re-evaluated at the end of each cycle. Digital Media and Flynow were further along in adopting agile while State University was still in an early phase of transitioning to agile.

As these software development groups were transitioning to the agile development approach, the adoption of wikis had begun as grassroots efforts in their organizations. Software developers were the early adopters who set up a wiki for their internal use. Eventually, the wikis caught the attention from other groups or departments, and received organizational support. Though the use of wikis was not mandatory, many software development project teams chose the wiki as their project information repository.

5.2.2 Work domain: Five software design projects

Similarities of software design projects are identified with the means-ends structure of the projects as shown in Table 5-2, which represent a typical software design and development project in a large organization.

Table 5-2: Similarities in the means-ends structure

Goals	To design and develop a usable web application
Constraints	Organizational long-term goal, rules and regulations, stakeholders, schedule, existing systems/applications supported by their organizations
Priorities	Delivering the application on time, following the agile approach, working successfully as a team
Functions	Planning, requirements gathering, design, development, testing, release
Physical processes	Scoping the project, scheduling, having project meetings, creating functional requirements, developing design specifications, coding, conducting usability studies, developing test plans and/or test scripts, filing and fixing bugs, releasing the software
Resources (Tools)	Whiteboard, wiki, email, a file-sharing server, bug tracking system, SVN servers, meeting rooms, a big screen display

The final products of Project A and B were Web applications, which were part of a suite of Web-based communication and collaboration applications designed for students, faculty, and staff, while those of Project C, D, and E were launched to the company websites. While the final products were different for the five projects, their general goals were similar, which was to develop a web-based application used by customers. Project A, C, and D were developing a new web-based application while Project B and E were enhancing an existing web-based application. The five projects had similar constraints that they were constrained by schedule, stakeholders, rules and regulations, and existing systems/applications supported by their organizations.

Their general functions were similar in that both teams had planning, requirements, design, development, testing, and release, which were quite generic for any software design and development project. However, work activities were different depending on the project's specific goals, constraints as well as the nature of the projects. For example, Project A was conceptual-design oriented, thus Team A conducted extensive user research to gather user requirements. The other four projects were more development focused; therefore, their project teams did not spend much effort on user research activities. For Project C, D, and E, the majority of project activities were developing and testing code, and collaborating with external stakeholders since these three projects involved implementing a third-party system.

Resources, in this case defined as the tools project teams used to support their work processes, were similar across all projects. Those tools include email, file-sharing systems, wiki, chat program, and bug tracking system.

5.2.3 Social organization: the team's structure and collaboration

Social organization is described in terms of the teams' structure and characteristics of their collaboration.

Team structure and responsibilities

All five project teams were cross-functional teams that consisted of members from different disciplines, many of which came from different organizational units both within and outside of the IT department. Although project members' titles differ from project to project, they can be grouped by their responsibilities and roles into five different disciplinary groups, which are business, user research, design, software engineering, and domain experts, as shown in Table 5-3.

Table 5-3: Team structure

Disciplines	State University		Flynow		Digital Media
	A	B	C	D	E
Management					
• Project manager	1	1	1	1	1
• Product manager			1	1	
User Research					
• User researcher	2				
• Usability researcher					1
Design					
• Interaction designer	3	2	1		
• User Interface designer					1
Software engineering					
• Software developer	2	2	4		3
• QA engineer	1	1			4
• QA systems analyst			3	1	
Domain experts					
• Pricing analyst			1		
• E-Commerce				2	
• Customer loyalty staff				1	
• Customer care staff				3	
• Linguist					1
Total	**9**	**6**	**11**	**9**	**11**

In general, project managers were responsible for managing the project team in order to ensure that the project was on track in terms of both timeline and outcome. For three projects (A, C, and E), the project managers also took additional responsibility to develop business or functional requirements. User researchers were responsible for user need-gathering activities at the beginning of the project as well as usability testing activities during the design phase. However, when there were no researchers in a project team, designers often took responsibility for usability testing. Designers were responsible for all design activities including from wireframing to hifidelity prototype. The designers were expected to develop design specifications, which outlined software interaction and interface.

After the design specifications were created, software developers were responsible for creating technical specifications and developing the software while QA engineers were responsible for testing the software. For Project A and B, designers also worked with software developers on front-end development. When needed, domain experts worked with other team members to ensure that the software worked correctly in specific areas. Roles and responsibilities for team members are summarized in Table 5-4.

Table 5-4: Roles and responsibilities

Roles	Responsibilities
Project manager	Project schedule, team coordination and communication, business or functional requirements
User researcher	Needs assessment, usability testing
Designer	Interaction design, user interface design, front-end development
Software developer	Front-end development, back-end development
QA engineer	Testing, maintenance
Domain expert	Domain-specific inquiry

Nature of collaboration

With an agile approach, more specifically Scrum development methodology, all five project teams had frequent and open communication among team members throughout the project cycle. However, interdisciplinary collaboration that happened during the design process comprised different involvement from each discipline. Figure 5-6 illustrates the intensity of collaboration across disciplines, where the heavier lines designate more intense collaboration. Project managers interacted with team members from all disciplines throughout the project cycle. Designers worked with the management group and the user research group to transform the business/functional requirements to design specifications. They also interacted with the software engineering group to ensure that the resulting code accurately reflected the design specifications.

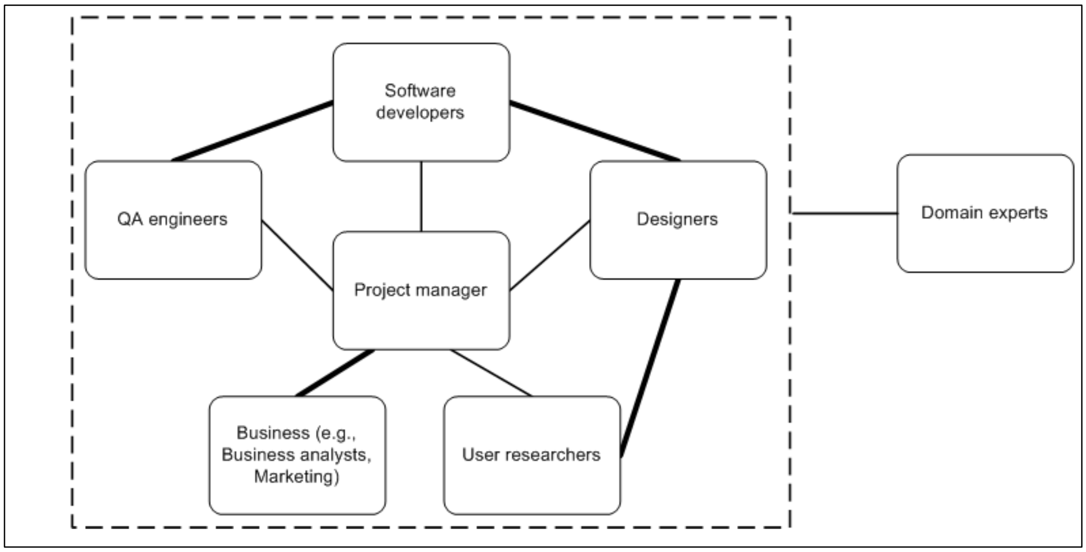


Figure 5-6: Collaboration across disciplines

Based on the observation and interview data, I observed two types of interdisciplinary collaboration: contributing and challenging. First, the project members contributed their discipline-specific knowledge when collaborating on tasks that required knowledge from multiple disciplines. For example, the Project C's use cases were created in a meeting in which a project manager and representatives of each discipline group attended. Second, the project team members often challenged each others' work through reviewing and commenting, particularly in a project meeting. They challenged each other with their discipline-specific knowledge. For example, developers would point out technical problem with the proposed design; designers would bring up usability issues of the proposed technical solution. Challenging collaboration was observed during (1) project meetings, (2) design discussions, and (3) bug-fixing activities.

5.2.4 Activity: Information sharing through project wiki sites

Table 5-5 presents a comparison of how software design teams used project wiki sites to share information. Although all project teams used a project wiki site as their central information repository, their use differed greatly from being merely an access point to project-related information stored in various tools to being a project collaborative space for team members.

Table 5-5: Project wiki site comparison

Project	Size (pages)	Platform	Purpose	Major contributors
A	76	MediaWiki	Collaborative space	Designers
B	45	MediaWiki	Publishing platform	Designers
C	7	OpenWiki	Access portal	Project manager
D	18	OpenWiki	Access portal	Project manager
E	3	MediaWiki	Publishing platform	Project manager

A wiki site is an access portal. Two teams, Project C and D, used the project wiki sites primarily as a portal to access project information. Because the Flynow's organizational policy did not allow files to be uploaded to the wiki server, these teams created links to project documents stored in the file-sharing server. The teams found that the project wiki sites provided easy access to project information because the project wiki sites provide project members to a central access point to various types of information including files stored in a file-sharing server, URLs of external Web resources, and wiki documents. In addition, with wikis, project members could give descriptive labels to shared information to help the teams easily find information.

"We have a file server that we actually put the documents [on] – we organize them in a folder for the project, so yeah, the wiki becomes this portal for linking." – C:Project manager

A wiki site is a publishing platform. With the project wiki sites, the project teams could easily publish their work, and share information across disciplines more efficiently. Project B and E used the project wiki sites as a way to disseminate information to the entire team as well as to project stakeholders.

"[The wiki] makes it easier for everybody to publish content on the web in a web-readable format." – B:Designer 1

"It's a very quick and easy way to get information up on a website, or compile anything, or to even publish information kind of in a spreadsheet and then throw that on the wiki so people [are] gonna look at. So it seems to integrate really well for being a consistent source of information that's constantly referenced and kept up to date. And it's very quick just to make a change, and then it's out there." – E:QA engineer

A wiki site is a collaborative working space. The Project A team used the Project wiki site, particularly by those who were more comfortable with the technology, to collaborate on design tasks, particularly creating information artifacts. Many wiki pages were co-authored by project members from different disciplines.

"Just really to use it as a place to do collaborative work so you know work on compiling documents collaboratively I guess a major use of it." – A:Designer 2

5.2.5 Actors: Team members

This analysis of actors aims to understand actors' resources and values that can influence the way they used project wiki sites to share information.

Team member profiles

Team member profiles were created from the members of all five project teams. The profiles include age range, gender, educational background, working experience, technical skills, and prior wiki experience, is summarized and discussed as follows.

Age and gender: The team members of software design projects in the study are from 18 to mid-40s, with the majority (75%) falling in to the upper range of 26 years old and older (Table 5-6). The project managers are in the upper range while most designers and developers are in the lower range. There are slightly more male participants than female participants in most teams, except for Project D, which had 9 females and 3 males.

Table 5-6: Member age and gender

Age/Gender	Male	Female
18-25	6	2
26-35	8	5
36-45	5	6

Educational background: Project member's education ranges from a bachelor degree to a doctoral degree, with majority holding a bachelor degree. There were two researchers who had a doctoral degree. Educational background is aligned with tasks and skills required by the project roles. Managers had business or marketing degrees. All designers had an interaction design-related degree while software engineers had a degree in either computer science or related fields (computer science, informatics and mathematics).

Work experience: The time that team members had been at their organization varies from less than a year to 22 years (Table 5-7). The majority of participants had worked at their organization between 1-5 years. There were a few new recruits who had started working less than a year prior. Team members at Flynow had the highest work experience at the organization with an average of 8 years while team members at Digital Media had the least work experience with an average of 2 years.

Table 5-7: Work experience at organizations

Work experience at the organization	**Projects**				
	A	**B**	**C**	**D**	**E**
Less than a year	1	1	0	2	4
1-5 years	7	4	5	5	6
6-10 years	1	1	3	4	2
More than 10 years	0	0	3	3	0

Technical skills and wiki experience: Using the interview data, I grouped participants based on their technical skills and their wiki experience. The majority of participants felt comfortable with computers and technology. Thus, their technical skills ranged from medium to high. All of them used a computer and at least basic software applications (e.g., word processor, spreadsheet, and Internet browser) for their job. There were five participants (researchers and customer care staff) whose technical skills were low as their jobs did not require the use of computer other than basic software applications. On the other hand, the software developers and QA engineers had high technical skills because they had advanced technology skills such as database skills and programming skills.

The wiki experience in this study was rated based on participants' wiki experience prior to using a project wiki site. During interviews, participants were asked how much experience they had with wiki technology at the beginning of the project. Wiki experience rating was determined as follows: (1) If participants had no exposure to any wiki sites or little experience editing any wiki documents, I rated their wiki experience low; (2) If participants reported that they were familiar with wikis and had some experience editing wiki documents, I rated their wiki experience medium; and (3) If participants had considerable experience with wikis, both viewing and editing, I rated their wiki experience high.

Most participants reported that they did not use wiki outside of work, except for reading Wikipedia. Thus, their prior wiki experience came from their use of wikis at work. Since EdApp was the only group that had started using wikis consistently for software development projects two years before this study started, most team members had some experience using wikis in previous projects. There were three exceptions. Two participants, the project manager and the interaction designer, had just joined EdApp while another participant, a researcher, had never been involved in a software development project. Table 5-8 shows that team members with higher technical skills did not necessarily have more wiki experience. However, team members with higher technical skills were more likely to pick up wiki skills quickly. On the other hand, all team members with low technical skills did not have any prior experience with the wiki technology. Five participants had heard of or read Wikipedia, but had not contributed to any wiki sites.

Table 5-8: Technical skills and prior wiki experience

Technical skills	Prior wiki experience		
	Low	Medium	High
Low	8	0	0
Medium	9	7	3
High	7	12	6

User perception

Project wiki sites are valuable: Participants from all software design teams stated that the value of wiki was really about information stored in it. As a result, participants viewed the project wiki as a place to primarily find information, which was time-saving. In addition, some team members thought of wiki as an archival place where they would store information beyond the project life cycle.

> *"It's been good, just as a repository of all the information we need. Basically, any question I had for [the project], you can go to the Wiki and get an answer." – B:Developer2*

> *"I like how all of the information about their project is in one place. And then, I like how all of the information is right there – mock-ups, all those kinds of things – in one spot, so I don't have to ask several people for – it just saves a lot of time." – D:Customer care2*

> *"You know, it's really the fact that it is, it has been a greater success as a single source of information." – E:QA engineer*

Team members had an initial positive attitude toward wikis: Based on the interviews with project team members, I inferred that in general, the members from all project teams had positive attitude toward wikis. The project managers from the five projects reported that their project members were open to adopting the wiki technology. At the beginning, no one had a strong feeling against using wikis. The participants from State University and Digital Media reported that the wiki technology was the first tool that had received a group buy-in. There were only two participants from the customer care group in Project D whose initial attitude toward the wiki technology was negative at the beginning. However, once they were more familiar with the wiki technology, their attitude became more positive.

> *"It's funny. I thought I would have more like oh man this is frustrating. It's probably hard to press me on negative things about the wiki."- A:Designer2*

> *"I don't really have any complaints about using the Wiki at all. It's much better than what we had before." -B:Developer1*

> *"[The] wiki does pretty good for what I use it for."-B:Designer2*

> *"We found it very restrictive, very scary, very high — too much information, very intimidating. And now that we're used to them it's amazing. They've become our friend, our source of information, and just it makes our lives so much easier." – D:Customer care lead*

Despite early reluctance, participants with less technical skill (i.e. non developers) and less wiki experience were eager to learn. Most participants reported more positive experiences than negative experiences. Some participants could not point out what they did not like about the wiki at all. Interestingly, participants were very forgiving of the wiki. Many participants had complained how the wiki did not work the way they had expected, but they ended the conversation by defending the wiki with a statement similar to *"it's nothing against the wiki"* or *"I can't say that's a complaint."*

Wikis are easy to use, flexible, and open: The members of all five project teams thought of wiki technology as easy to use for both sides of sharing: putting information into the project wiki site, and getting shared information from the project wiki site. Ease of use contributed significantly to their positive experience of the project wiki site. Twenty-two participants associated the wiki's ease of use to its editing functionality due to its simple markup syntax, which also made it easy to add, edit and update shared information. Additionally, fifteen participants explicitly stated that they liked the open-editing feature and found this feature to be beneficial to information sharing. The project wiki sites were linked to their organizational wikis, which were open to everyone in the organization to view and edit.

> "*I think the ease of use of the wiki is really great." – A:Project manager*
>
> "*I like the fact that you can edit so easily." – A:Developer 2*
>
> *"I like it that it's easy, you just click my information. It's very easy to add information to, as far as moving documents on to." – D:Customer care lead*
>
> *"I think the biggest thing is it's just simple." – E:Project manager*

In addition, despite of its poor search functionality, many participants found that the wiki project site was easy to navigate to needed information, particularly when they compared it to other information sharing tools they had used such as the file-sharing server, Microsoft SharePoint, and email. Many participants said that it had been much easier for them to find and make sense of information shared by other disciplinary groups than to go through many folders in the file-sharing server. Participants who had experience using other information sharing tools thought that the wiki was better because it was more flexible. With wikis, participants felt that: they had more control; they could create and share different types of content; they could access shared information more easily; and they could manage shared information better.

> *"My response to that is with [the] wiki I feel like I have more control over the land. With SharePoint, I'm literally confined to this left bar [...] whereas in [the] wiki the project space kind of evolves over time. There [is] something that came with it and I am able to add some white space to keep it simple. It's just more control." – A:Project manager*

> *"It's cleaner than going to a file server and trying to look in that folder and search for stuff. The file server's ugly. You've got to find the name of the file. This way, you can link it and call it whatever you want." – C:Project manager*

> *"Yet, we can work like we are, like we just talked about it because they can update something, I can look at it, I can see it, immediately, I don't have to wait for an email, I don't have to slog through 1,600 emails and try to find somebody's email to me. – D:Customer care lead*

The use of wikis encouraged more sharing across disciplinary groups: Participants believed that the project wiki sites have encouraged the project team members to share more information, particularly across disciplinary groups. Some participants explicitly said that the wiki technology had made them more willing to share information, while others felt that they were exposed to more information, and that the shared information was more visible and easily accessible via the project wiki sites.

"I think the real difference was just the ease and willingness to share information. Like people wanted to share information too but there were always taking steps like 'I shared that' or 'I put it on the file server'." – A:Developer1

"I think it's made it so that I'm more willing to contribute content to things. It's made it so that I'm more willing to use what's here, because I imagine other people are more willing to view content too." – B:Developer2

"I think it just makes the whole design process more visible to the development, you know. I mean, I think that, you know, we have the one-pager and the Mantra up, they see all the little incremental sort of steps that we're taking toward the design." – A:Designer1

The project teams at State University recorded and shared more ad-hoc information after they started using the wiki as one developer explained, *"I think what [the old intranet system] did, was it made the barrier to entry sufficiently high, so that you would never record an ad-hoc conversation [...] whereas in the wiki you get much closer to trying to capture, even trivial artifacts, just thoughts and fragments. Even if those thoughts don't make it onto - you create a page at Wiki you can abandon it."* A designer described that the project wiki site was "*a good place to put down ideas that you have to make sure that it's documented somewhere and you can always come back to it. But evidently, you can't always come back to it.*

5.3 Context Summary

This chapter described individual contexts (five projects in three organizations) that together characterize a common context for the use of the wiki technology studied in this dissertation. The common context is discussed in CWA analysis dimensions as summarized in Table 5-9. In sum, wiki technology was used as a central information sharing tool for a web application design and development project, situated in an IT department of large organization that recently adopted an agile development approach. The project team consisted of members from four or more disciplines including project management, business, research, design, software engineering, quality assurance engineering, and other domain experts. Team members differed greatly in age, educational background, work experience, technical skills, and wiki skills. However, most of the team members had a positive impression and experience using the project wiki sites.

The common context is discussed in CWA analysis dimensions as summarized in Table 5-9. These dimensions imposed constraints that shaped how project wiki sites could be used. First, the constraints imposed by the work environment included the technology use policy, document access policy, and information systems currently used in the organization. These constraints influenced what and how information could be shared on project wiki sites. For example, Flynow did not allow wiki users to upload files to the wiki server, so wiki users had to create links to files stored in a file-sharing server. Second, the work domain and activity dimensions influenced how project teams used and structured project wiki sites. That is, project team structured the project wiki sites to support their design and development process. Third, team structure and the nature of collaboration influenced how individual project members used project wiki sites. For example, project managers used the project wiki sites to monitor a project progress, thus they frequently accessed the project wiki sites. Designers, on the other hand, used the project wiki sites to created and stored design artifacts; therefore, the designers appeared to make most contribution to the project wiki sites. Lastly, actor's resources and values influenced how much individual project members were willing and able to use project wiki sites. The project members who had more technical skills and prior wiki experience would felt more comfortable using project wiki sites. Moreover, project members with positive attitudes toward wikis were willing to learn and use project wiki sites.

Table 5-9: Common context for the use of wiki technology

Context dimension	Characteristics
Work environment	IT department in a large organization (1,000+ employees) that recently adopted the agile software development
Work domain	Web application design and development projects
Social organization	*Team structure:* Project manager, user researchers, designers, software developers, QA engineers, and domain experts *Collaboration:* Contributing each other's work; Challenging each other's work
Activity	Using the project wiki site as: (1) a portal to access project information; (2) a publishing platform to disseminate project information; and (3) a collaborative space to collaborate project tasks
Actors	Profiles: wide age range (20 - 50 years old), bachelor's degree to PhD in various fields including computer science, business administration, informatics, and other industry-related fields; varied in working experience (6 months to 22 years); mostly tech-savvy with various wiki experience. *Perceptions:* Project wiki sites are valuable; Team members had initial positive attitude toward wikis; Wiki is easy to use, flexible, and open; The use of wikis encouraged more sharing across disciplinary groups.

CHAPTER 6: INFORMATION SHARING THROUGH PROJECT WIKI SITES

This chapter presents the findings of how software design and development teams in three organizations used wikis to support information sharing across disciplinary groups during the design process. First, I discuss why the software design teams chose to adopt wiki technology and how they envisioned their teams using the wiki technology. Second, I describe what information artifacts the software design teams shared on their project wiki sites. Third, I discuss the information sharing needs that the design teams tried to meet when sharing those information artifacts on the project wiki sites. Finally, I discuss what the software design teams actually did when sharing information on the project wiki sites to meet their information sharing needs. These findings are shaped by the 4-T framework and CWA framework. CWA's activity analysis revealed why, what, and how wikis were used to share information in the studied context whereas the 4-T framework was used to frame the findings on information sharing processes.

6.1 Why a Wiki?

The interview data revealed that the project teams expected the project wiki sites to be a central information-sharing place where they could easily find all information they needed in order to complete the projects. The project members expected to have project artifacts posted on the project wiki sites, as *"Did you put it on the wiki?"* or *"Can you post it on the wiki?"* expressions came up throughout my participant observation in Project A and Project B at the State University.

"Basically it'd be one place that the team would go to for all information generated about the project and artifacts we generated. Just having one place to go." –A:Developer2

"So, I gave them a centralized point to easily navigate information and pull it together."- D:Project manager

"There're use cases, three mock ups, a task manager tool, a bug list, list of bugs and their resolution from the vendor, and actually, there's another tool that we used for bug tracking, and all of those things are linked to our wiki." – C Project manager

"I want to make sure it's linked [on the wiki] so people can find." – E Lead developer/ Scrum master

A project wiki site served as a centralized information space for heterogeneous information items used by the teams. Even though some information artifacts were not suitable for storing on the project wiki sites, for example, bug reports and codes, many participants expected to be able to use the project wiki sites as a single access point to such information. Table 6-1 demonstrates how the project wiki sites were used as a central information sharing tool to link information from other places into one place for Project A and Project B.

Table 6-1: Demonstration of a project wiki site being one place for all

Tools	Project A	Project B
Bug tracking system	• 2 saved search bug lists linked from the project wiki home page • 8 bug reports copied to the template to-do list wiki page • 8 bugs referenced in the usability findings	• 2 saved search bug lists linked from the project wiki home page • 3 bug reports linked from the project wiki • 12 bug reports copied to the functional requirements wiki page
Email	• 8 email messages were copied and posted on wiki. • 21 email messages contained a link to wiki pages. • Files and links shared through email were posted to the wiki.	• 3 email messages were copied and posted on wiki.
File-sharing server	• 2 links to the project folder on the file server was posted on the homepage. • 3 files in the file server was linked from the wiki • # of files stored in the server that were posted on the wiki	• 2 links to the project folder on the file server was posted on the homepage. • 5 files in the file server were linked from the wiki. • 3 subfolders in the project folder were linked from the wiki.
Request Tracker	None.	• 14 of request tickets (RT number and summary) copied to the project wiki. • 2 links to RT were posted on the project wiki.
Other online information spaces	• Material and information on BackpackIt copied to the project wiki site • 2 links to the research online file sharing space	None.
Whiteboard	• 30 photos of whiteboard sketches were posted on the meeting notes page. • 9 meeting notes contained transcript of whiteboard sketches. • 3 instances were observed when wiki content was written on the whiteboard to continue discussion.	None.

6.2 Information Artifacts Shared on the Project Wiki Sites

The five software design teams shared many kinds of information artifacts between disciplinary groups throughout the design process. The information items shared by different disciplines differed in content, form and format, and were often created by different tools. The team members had to make a decision on what to share on the project wiki sites. When asked about how they decided what information artifacts to put on the project wiki, most team members agreed that all project-related information that needed to be shared with the team should live on the project wiki. Participants expressed confidentiality as criteria to decide whether a document should live on project wiki sites. At Flynow, in particular, the project team members were more concerned with avoiding putting confidential documents (such as federal regulations) on the project wiki site.

> *"That was probably one of the problems with us adopting the wiki too was making that choice all the time and saying, is this something that should go on the wiki?" – D:Customer care lead*

> *"I do feel like putting everything on the wiki is important but structuring it in a way that's not overwhelming. […] Really, every information, every piece of notes is great to be up on the wiki." – A:Designer3*

> *"Well, what wouldn't you put out on the wiki?" I'll tell you the answer: everything goes on the wiki. I pretty much—if I use the wiki I throw everything on the wiki. But if you have confidential documents, you don't put them on the wiki." – D:Project manager*

Types of information artifacts

A common theme coming from the interviews with participants of all five software design projects was that ***everything should be shared on the project wiki sites***. Thus, the information artifacts shared on the project wiki sites provide a good representation of the types of information artifacts commonly shared during the design process. I analyzed more than a hundred artifacts that were shared on the five project wiki sites (as shown in Appendix D), and developed three main categories: framing artifacts, design artifacts, and procedural artifacts.

Framing artifacts are information items shared in order to help define a design problem, build a shared understanding of design problems, or guide design decisions. Business requirements or functional requirements were reported as the most important framing artifact that helped them understand what their project team was trying to design. Designers used this information to guide their design while developers used this information to plan the backend development. Different design projects had different documents to account for these requirements. Project A and Project B had "functional requirements," Project C and Project D had "business rules" and "use cases," and project E had "business architecture."

Design artifacts are information items shared in order to represent design ideas and design solutions. These information items are usually produced as output of design tasks, and are sometimes used as input for another task. The most important design artifact reported by all teams was a design specifications document (also known as a blueprint by the Project Team C), which was an important design artifact that was used primarily by software developers and QA engineers.

Procedural artifacts are information items shared to support tasks and collaboration. Shared task lists and bug reports were important procedural artifacts. All project teams had some kinds of task lists that they used to keep track of certain tasks or overall project progress. Bug reports were important for most disciplines (except for the user research), particularly during a later phase of the project.

6.3 Information Sharing Needs

To better understand the ways in which software design teams used the wiki technology to share information, it is important to understand their information sharing needs. Participants from all projects reported that they shared information artifacts through the project wiki sites because of the following motivations:

To provide access to project-related information: Participants often described that they put documents or information items into their project wiki sites so that other project members could access those documents as necessary. For some information artifacts, participants did not anticipate that project members from different disciplinary groups would view or use the information that they shared. For example, the QA group in Project C put many testing-related documents in a project folder in the file-sharing server as well as posted them on the project wiki; however, they were for their own use as one project manager described, *"QA uses it. [...] That's the testing area. There's just a bunch of miscellaneous documents we put on there in case anybody needed to look at them."*

To keep project members up to date: Design project team members put information on the project wiki to keep everyone in the project up to date with what was currently going on, especially in different discipline groups. Even though information was not needed directly for specific tasks, it was useful as a means of keeping track of the project in order to know where things were in the project and to know what each other were doing. Sharing information via different channels helped foster such communication. All participants reported the need to share information on the project wiki sites as a way to keep project members as well as stakeholders up to date. A few participants personally liked to know what other project members were up to.

> *"I like to look and see where people are on a thing. I don't feel at least in this group it's a judgment. It's very much just to help for their own knowledge." – A:Designer2*

> *"So we have to keep remembering to put it in there so that everybody over here could view what we were doing, what the latest documents were." – D:Customer care lead*

To build a shared understanding: Shared understanding was crucial to software design and development projects as mentioned by many participants and also observed during the study. A designer explained the need for building a shared understanding as he said, *"Definitely what I need is for them to understand what we're coming from. Because a lot of time, just because [we're on] different teams, it's hard to. I mean we just don't speak the same language – different backgrounds, different ways of understanding, the concerns at any given point."* Participants from the five project teams stated that they shared information artifacts on the project wiki site to develop a shared understanding among team members. Particularly in an early phase of a design project, the project members spent more time defining, scoping the problem space as well as figuring out design solutions. During this time, the project members shared information to help build a shared understanding of the design problem.

"Oh, here's the one-pager, I referred to it at least for a while until I internalized it enough that I could make sure that all our decisions were within that. I think it being in here, I can't speak for the whole group but I think having that there was good for group understanding." – A:Desinger2

To complete a task: The majority of participants reported that they either shared information because other project members needed such information to do their jobs or needed information shared by others to do their jobs. All participants reported that they shared information artifacts on the project wiki sites because of information anticipated to be needed by other project team members. Information shared across disciplines was used as input for particular tasks as illustrated in Table 6-2.

Table 6-2: Shared information needed for completing tasks

Information	Shared by	Needed by	For task
Project scope/ Project charter Project schedule	Project manager	All	Planning their tasks
Business rules/ Use cases/ Functional requirements	Product manager Business analyst	Designers Developers QA engineers	Designing UI and interaction flow Designing software architecture design Developing test cases
Technical constraints	Developers	Designers	Designing UI and interaction flow
Design specifications	Designers	Developers	Coding the software

Anticipated information need was identified in various ways, such as project deliverables listed in the project definition document, experience in previous projects, and links in the project wiki page template. For example, designers created and shared the design specifications on the project wiki sites because it was a project deliverable and was needed by developers in order to develop designed applications.

> "*I know the developers automatically come to look at the spec. So it's kind of a clear-cut way for them to know where the information is." – B:Designer 1*

Some participants reported that they posted information artifacts on the project wiki sites in response to explicit requests from others. For example, a developer posted statistic data on a project wiki site because it was requested by one of the designers on the team.

> *"The things I put on for the [wiki] would be this, if the design team asked me for statistics about how people have been doing. I would have written up and recorded here. ... For example, this type of thing: how many people have done this in the tool, how many people have done that so forth. I would fill that out and make a page for it." – A:Developer2*

To complete their own tasks, as lot of design work involves reflection and negotiation among project members, the project team also shared information frequently to receive feedback from the project members. This occurred more in Project A than in other projects because there was more initial design work in Project A that needed a lot of reflection and feedback across disciplines. Information artifacts that were shared on project wiki sites to receive feedback include: scenario, functional requirements, one-pager mantra, screen flow diagrams and wireframes for Project A; functional requirements and design specifications from Project B; use cases for Project C; and test cases for Project E.

6.4 Information Sharing Processes

To meet their information sharing needs, the design teams engaged in multiple levels of sharing identified in the 4-T framework, namely materialistic sharing, syntactic sharing, semantic sharing, and pragmatic sharing as shown in Table 6-3. In this section, I discuss how the design teams used the project wiki sites to achieve each sharing level.

Table 6-3: Mapping information sharing needs to the 4-T framework

Information sharing needs	Information sharing levels
To provide access to project information	Materialistic sharing (accessible)
To keep project members up to date	Syntactic sharing (viewable)
To build shared understanding	Semantic sharing (understandable)
To complete a task	Pragmatic sharing (usable)

6.4.1 Materialistic sharing: Transport

Sharing artifacts may involve multiple levels of sharing as presented above in Table 6-3. Materialistic sharing is the first step of sharing that concerned information access, that is, how to provide other project members access to shared information artifacts. Software design teams used project wiki sites to support transporting information artifacts across disciplines by creating a project wiki site, adding information artifacts in the project wiki site, linking to shared information artifacts, organizing the project wiki site, and pushing information artifacts to the project members.

Creating a project wiki site. Creating a project wiki site could be as easy as creating a new wiki page within an organizational wiki space (at State University and Digital Media) or more complicated if requiring a request to the central IT department (at Flynow). All project wiki sites consisted of a project homepage, which served as portal page containing links to information in various forms: wiki content pages, image files, links to wiki pages outside of the project wiki, or external web pages. Project homepages were structured to reflect their cross-functional teams and different project phases. EdApp's project wiki homepage consisted of seven sections: Working Documents + Meeting Notes, Project Management, Research, Design, Development, Technical/Marketing Writing, and QA (Figure 6-1). Each section was filled with content from activities for each phase. For example, the research section consisted of links for competitor analysis, a literature review, log data, user research, and a summary of research findings.

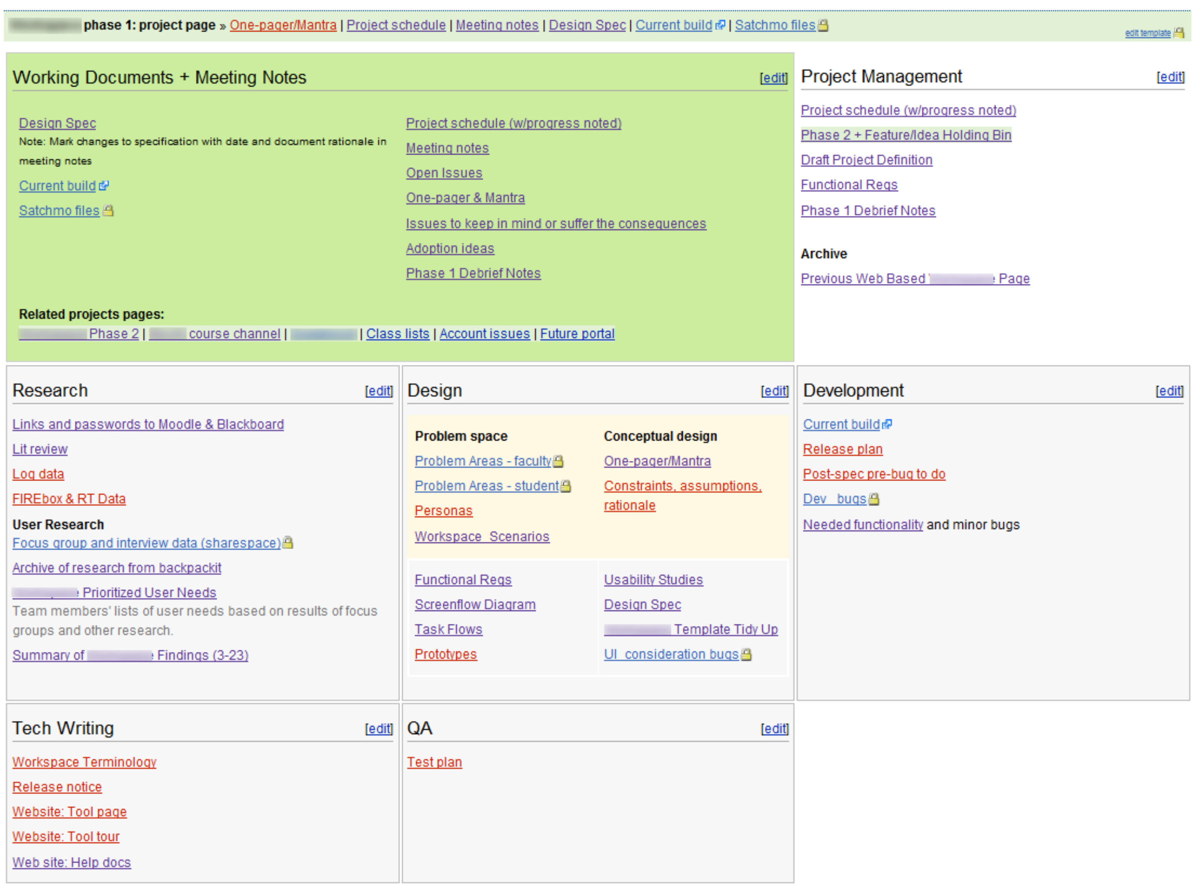


Figure 6-1: Project A wiki homepage after the project was completed

The project wiki homepages of Project C and Project D contained sections representing project activities such as Project Management, Technical Specification, and Testing. Similarly, the Project E's wiki homepage at Digital Media was structured to reflect project phases as the project manager explained, *"It's not the actual table of content but it's an easy way for you to see the planning analysis, design, the standard project phases are kind of broken out on the wiki page. And you can jump to whatever section."* Several participants felt that a project homepage helped them find information more easily and provided an overview of where the project was.

> *"I mean I just felt like in every project you never knew where to look to find something. [With the project page template] I think it's just a little easier to find some like documents." – A:Designer 2*

> *This is really my orientation of what's happening on a project team and what elements are, how they work together over time." – A:Researcher1*

However, having too much structure on the project wiki site could make it difficult to find information. While most project members appreciated the idea of using a pre-defined structure for the project homepage, and agreed that it was useful, some participants thought that the homepage was too busy with redundant information, information they did not need, or placeholders for future content. Some participants did not understand the headings and labels used in the project homepages; thus, they could not find information easily.

> *"Yeah, I like to see it like ideally it should be very clean. So even though a small link like project schedule it shouldn't be [in the working documents area]. If other people use it here, that's fine. But if we all agree that we don't need it there, we should just leave it*

under project management. So, slight things like that. I like to see the QA test plan here. Want to see a lot of these actually get used." – A:Designer 2

"There's nothing worse than a bunch of links that go nowhere. Another bad thing is when we sit down with wiki pages and start filling them out as if we can predict the future. But we don't know what it's about. All we need to do is figure out what it is and come back to the wiki and capture so we can have it ourselves. So we start ahead of ourselves and we quickly fall behind because the way in which we got ahead of ourselves wasn't useful. There's no value." – E:Developer manager

Adding information artifacts in the project wiki site. Once the project wiki site was created, the next step was to make shared information artifacts accessible from the project wiki site. There were two ways to add information artifacts to the project wiki site. First, the project teams created information artifacts in the project wiki site by creating a new wiki page. Design teams reported that when they created a new wiki page, sometimes they would copy a page layout of a wiki page with similar content. For example, all interview notes from the project A's first usability study used the same sections with same headings, which are "Initial Reaction", "Read through", "Tasks", "Post-study questions", and "Terminology."

"When I go in there the first time, usually what I'll do is go over to another wiki and copy a little bit, including a link." – C:Project manager

Creating a new wiki page was easy as explained by a QA engineer at Digital Media, *"you go to the page where you wanna link from, and type in this new page name and that'll create a link, and then there's your new page."* However, deleting a wiki page was not as easy as creating it.

In fact, the delete feature was restricted to administrators in all three organizations. Second, the project teams added information artifacts to the project wiki site by uploading files to the project wiki site. Many information artifacts were created as document files such as word files, presentation files, and spreadsheet files.

The decision of how to add information on the project wiki was largely affected by individual preferences, document types, and organizational practices. For instance, organizational policy at Flynow had prevented file uploads to project wiki sites as it was stated clearly on the organizational wiki index page on that they did not allow wiki users to upload files to the wiki server. Unlike EdApp where there was no file upload restriction, over a hundred image files were uploaded to each project wiki at EdApp. Most participants agreed that if an information artifact required advanced or specific formatting, it would be created outside of the project wiki, and uploaded as a document file to the project wiki site.

> *"If it's just text information I usually put on the wiki, if formatting becomes very difficult like user requirements. The reason that I used Word is because formatting gets very important and we usually have outlining and organizing information so that would be easier to do in word." – B:Project manager*

> *"The one thing that I think is our lessons learned - we will not use [the wiki] again to document and edit use cases on the wiki itself. We'll put them in a different document and link them in. Adding is just a nightmare. We wasted a lot of time trying to get them formatted right when it'd just be easier to put them in Word and then just link that document." – C:Project manager*

Linking shared information artifacts to the project wiki site. Linking is one of wiki's key features, which allows design project teams to build a relationship between shared artifacts. Once the information artifacts were added to the project wiki site, the project team always linked them to one or more pages in the project wiki site. Most wiki pages that contained project-related information were linked directly from a project wiki homepage.

The project teams not only created links for shared information artifact added to the project wiki site, but they also created links for external resources and artifacts not stored on the project wiki site such as bug reports, code. Although many artifacts were already accessible to the project team members in various places (such as a project folder in a file-sharing server, a project SharePoint directory, and a bug tracking system), the project teams added links on the project wiki sites to provide easy access to those artifacts, as illustrated in Table 6-4.

Table 6-4: Links to information artifacts outside of the project wiki sites

Link types	Project A	Project B	Project C	Project D	Project E
Links to files in a file-sharing server	5	4	28	14	-
Links to a folder in a file-sharing server	1	2	2	7	-
Links to other sharing tools	8	12	2	12	2
Links to external websites	15	34	-	4	

Project Team A and B developed navigation templates as a way to provide access to information in the wiki. The navigation templates used by Project A and Project B were organizational navigation template, project navigation template, and document navigation template. The global navigation template contained links to the organizational wiki homepage and team wiki pages while the project navigation template contained links to anticipated frequently accessed project wiki pages, as shown in Figure 6-2.

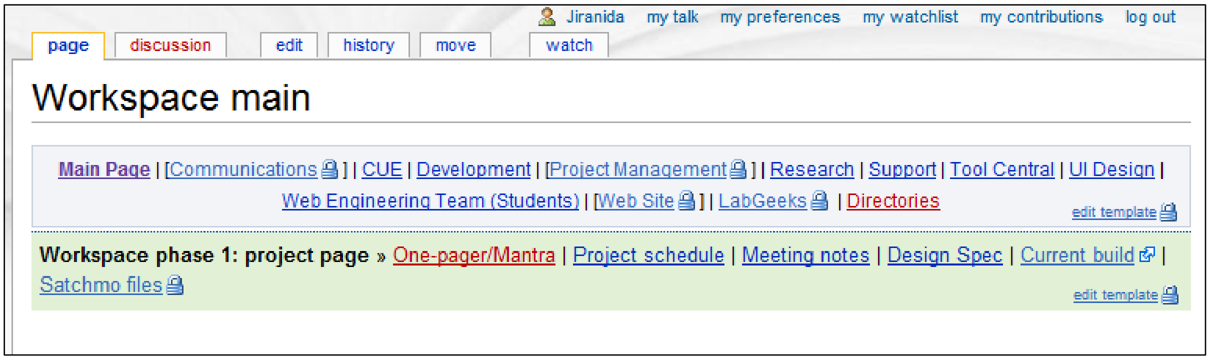


Figure 6-2: Navigation templates

Organizing the project wiki site. Wikis made it easy to share information in a central shared space; however, it was not always easy to find specific information. Often participants reported they could not remember exactly where a specific document was located on the project wiki sites. Some wiki pages were created but not linked to any other pages. These pages are called 'orphaned pages.' For example, there were 228 orphaned pages in the EdApp organizational wiki site, and the only way to find these pages was to search. Many participants reported that the search feature in their project wiki sites was not helpful in finding what they were looking for.

> *"I know that there are some pages that are out there with important information but people can't find them. The search feature isn't that good." – B:Project manager*

> *"The one thing that I really have found with the wiki is it's hard to search. In my opinion, the search is very poor. Unless you're watching changes every day, it's hard to find new information." – B:Developer 1*

“I've seen that happen occasionally where someone's, like, ‘I can't find this on the wiki,’ and maybe it's a name that doesn't make sense, but you have to look at your team. I've seen it happen.” – C:Project manager

To help project members find artifacts shared in the project wiki sites, the design teams organized the project wiki sites by using descriptive link labels, developing naming conventions for wiki pages and files, assigning categories to wiki pages, and creating a placeholder links for future content.

Common to all five projects is that their project wiki sites contained many links including links to wiki pages, files, Web sites, and tools. A label is how a link is displayed on a wiki page. When a link is added to a wiki page, it is given a default label of a wiki page title, a URL address, or a file name depending on the type of linked artifacts. However, with the wiki technology, any link could have a custom label using wiki markup. Thus, the project teams had flexibility on naming and labeling information items in the file-sharing server, as stated by C:Project manager, “You can put a user friendly name on it. I don’t have to look for the exact name of the document. It gives it an easy name. It’s really just a portal.” The project teams often gave descriptive labels to the links created in their project wiki sites. These descriptive labels helped the team members find shared artifacts more easily. Table 6-5 shows examples of descriptive link labels observed on software project wiki sites.

Table 6-5: Descriptive link labels

Wiki page title/File name	Link labels
ProjectA IssuesGlobal	Issues to keep in mind or suffer the consequences
Project A Great ideas	Phase 2+Feature/Idea Holding Bin
Project B v1 list	Old list of items required for v1 release
2006111ESRR.ppt	TRRS Re-Shop Reissue Presentation
TestingScenarioFailureLog.xls	Log of [Project C] Test Scenario Failures
Iteration Schedule Detailed v3.xls	Latest Iteration Schedule
[Project D] Charter_v3.doc	Charter V.3 - Approved & Update on 3/6

In a file-sharing server, project artifacts were organized in project folders and sub folders, which made it easy to find artifacts in a specific project by browsing or searching within a project folder. In most wiki platforms, such as those used in the three organizations, artifacts and wiki content are organized by users linking them to a project wiki homepage. Thus, it was more difficult to find artifacts in a particular project without browsing through the project wiki site. To address this issue, two project teams (Project Team A and Project Team B) attempted to develop naming conventions for wiki pages. For example, a designer in Project B had his naming scheme for wiki page titles that started with a project name as he said *"it's a little more human-readable."*

Another way to organize information shared on the project wiki sites is to assign a page category. Many wiki platforms offer the functionality to tag wiki pages. With MediaWiki (used at State University and Digital Media), the project teams could assign multiple page categories to any wiki pages. A page category is a way to tag wiki pages, which is then used to generate a page index. The page index is another way to access project wiki pages. The Project B team used the page category extensively as they created a "Project B" category and applied this to most wiki pages in their project wiki. The Project A team, however, did not make use of a page category. One designer on Project B said that though he had applied a project category to most wiki pages in their project wiki sites; he believed that his team members did not make use of the category page as he said *"...if you look at the category page for [project related content], they give you an index of every page that's in the category. [It] could be really useful if people used it more."*

Project team A and project team B created placeholder links for future content so that team members knew where to share information. The placeholder links were for not-yet-existed pages, but anticipated to be created in a near future. For example, the project homepages had a lot of links copied from the project wiki page template, and the usability study schedule page had links for individual interview notes.

Pushing information artifacts towards the project members. Having information artifacts accessible and viewable on the project wiki site did not guarantee that other project members would access them. Therefore, project members employed several methods to push the content towards the intended audience. Eight participants reported that they sent email project members with either the content in the email body, an attachment, or a link to the information item, asking the project members to review the shared content. Additionally, six participants reported that they often explained in email why and when they would like the content to be reviewed.

> *"So what we did before the — we have the test case review early on in the week, and prior to that, the person who wrote the test cases usually sends it out by email, the link to the wiki area they created. Ideally, all the team members would look at that and read it before." – E-Developer*

A project wiki was displayed on a big screen during a project meeting as a means of pushing shared information artifacts across disciplines. Team C and E had daily Scrum meetings, while the other teams also had regular project meetings throughout the course of their projects. For example, during one project meeting, the project manager logged into the project wiki at the beginning of a project meeting, and clicked on "a task manager" link to bring up a burn-down chart document, which was a graphical representation of work left to do. The project manager explained what was going on in terms of hours spent in the last sprint. Most project teams preferred to have the meeting notes up on a big display as someone was typing up to assure that other project members read the notes. In the past, two project managers used to email the notes to others, but they often got no response. Thus, they did not know whether the meeting notes were viewed or not.

> *"Before, I would send email with my notes out. Well, I still do that. Sending out an email with the notes as opposed to having them written during the meeting on the wiki. It was that the combination of the wiki and the big screen monitor. Because with the big screen monitor, we can review the document together, just [a] huge, huge benefit."- A:Project manager*

When participants were asked how they knew whether the information they shared on the wiki was viewed, many participants reported that they either knew from a face-to-face conversation, questions, and feedback on the shared information. The wiki log data showed that both project wiki sites were highly viewed when comparing the number of views to the number of edits (as shown in Figure 6-3), which indicates that the information was viewed through the project wiki sites. On average, wiki pages were viewed 62 times (excluding the project wiki homepages as outliers of 1,565 views and 915 views), and files uploaded to the project wiki sites were viewed 9.3 times. Highly viewed pages included project wiki homepages, meeting notes, design specifications pages, and usability study pages.

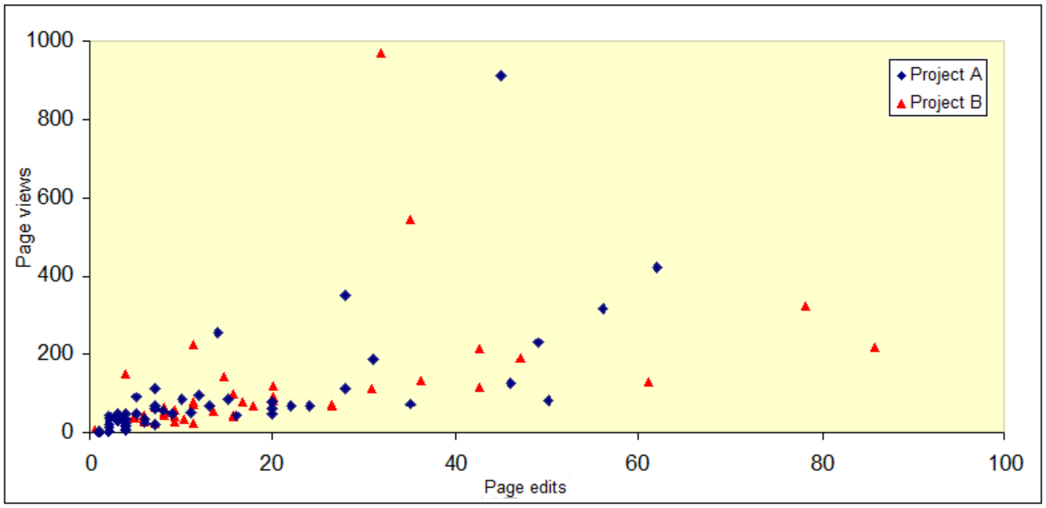


Figure 6-3: Page views and page edits of Project A and Project B

Some participants used ‘Recent Changes’ and ‘My Watchlist’ to monitor updates and changes on the project wiki sites. The Recent changes page is an automatically generated wiki page that tracks up to 500 recent changes in the entire wiki within the last 30 days, while the My Watchlist page is a personalized list of recent changes. The My Watchlist page allows participants to monitor wiki pages of their interest and to generate a list of only recent changes made to those pages. Figure 6-4 shows a screenshot of the Wikipedia’s Recent Changes page to illustrate what information is provided on Recent Changes.

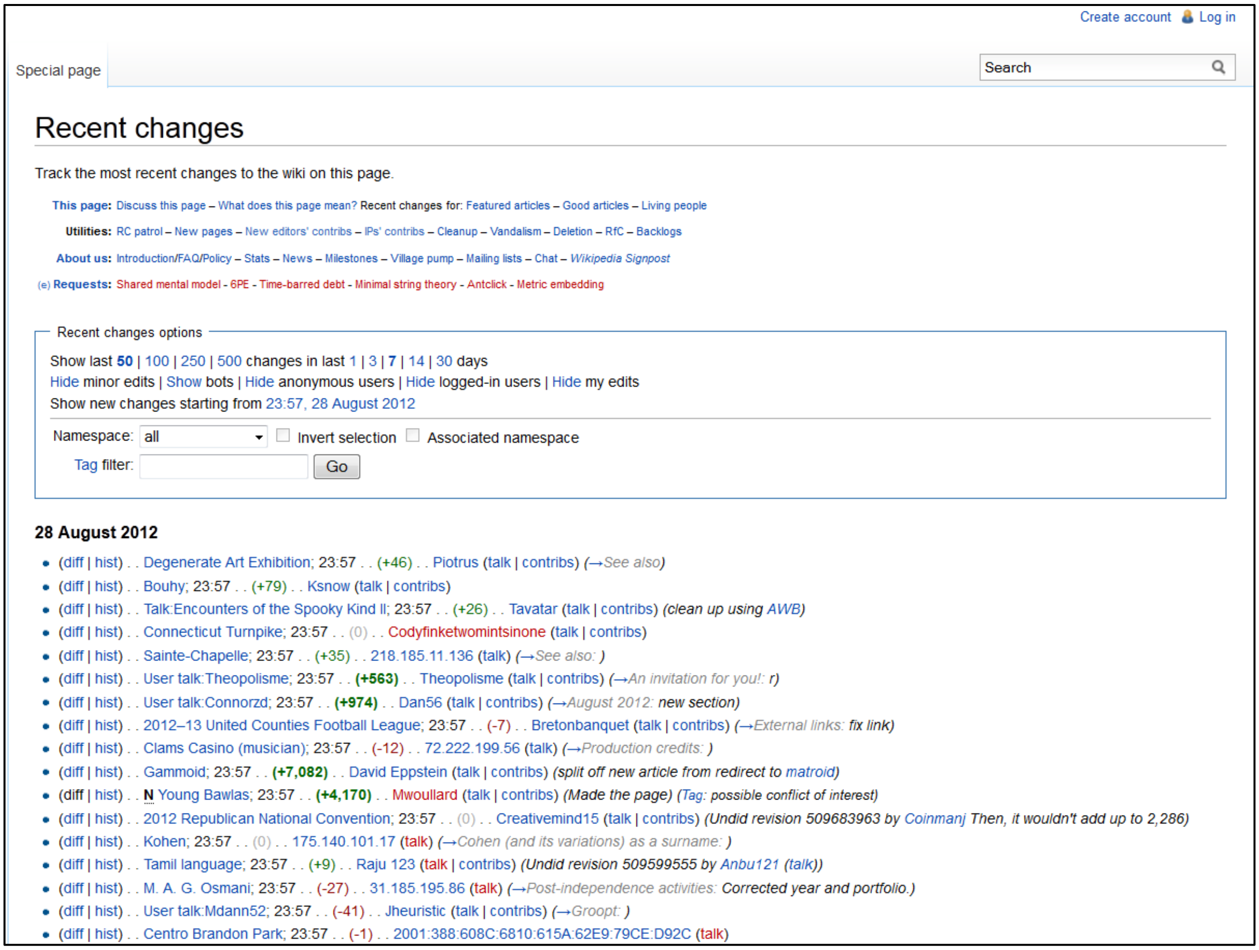


Figure 6-4: Recent Changes

Participants at State University felt that the Recent Changes page was too overwhelming because it included changes on all wiki pages in their organizational wiki site. Thus, it was difficult to use the Recent Changes page to keep track of only the wiki pages in the project wiki sites. On the other hand, My Watchlist is limited to changes made to their selected existing wiki pages; participants would miss changes made to other wiki pages as well as new wiki pages.

> *"I'm a big user of recent changes. That's how I use the Wiki. What I do I basically keep that page up. And I just refresh every now and then. … You just need to click on any*

one of these and it will give you the newest one. Yeah, you don't need to click on all of them. I find that helps me stay current and know which page to read." – A:Developer 2

"So it's kind of like, you gotta look at what's changed in the wiki. Some people – I know that you mentioned a meeting a while back – but in looking at recent changes in the wiki so that you can stay abreast of what's happening there. I do that for like a week." – B:Designer 2

"Occasionally if I'm trying to find a document that was put there recently, I will look at the recent changes, that also shows edits, but I don't use it that much." – C:Project manager

6.4.2 Syntactic sharing: Transfer

After having information artifacts that are accessible, the next level of sharing, which is syntactic sharing, concerns getting the content shared across disciplines. The design teams transferred an information artifact by making its content legible by project members from different disciplines. The design teams used project wiki sites to transfer information across disciplines by converting content to wiki-viewable format, formatting wiki content, and displaying viewable-size images.

Converting content to a wiki-viewable format. Both textual and graphical content could be easily viewable on wikis. The wiki technology provides a universal publishing platform for project members from various disciplines. That is, team members do not need a discipline-specific application in order to view wiki content. Participants from the five project teams reported that wiki content was more easily viewable because anyone with a web browser could access and view wiki content if it was text or images.

“The issue I have with Sharepoint is like email in terms of it’s a document that you have to open to actually consume the information. Whereas the wiki has a lot of information presented right away.” – E:Project manager

Therefore, some participants copied or converted content in other formats to wiki-viewable formats (from Photoshop, Illustrator, Visio diagram, or Microsoft Project files to image files, and from word document or SVN code to wiki text content). For example, the project manager from the Project E converted business roadmap documents into wiki content so that those documents could be viewed easily on the project wiki site.

Some document files were converted into image files because not all team members had software to open and view the original format. For example, task flow diagrams and screen flow diagrams were converted from a Visio file into an image file. The project schedule was converted from a Gantt chart, which was created in project management software, into an image file before posting it on the project wiki site. Table 6-6 shows examples of content that was copied or converted into wiki content collected from Project A and Project B wiki sites.

Table 6-6: Examples of content copied or converted to wiki formats

Information items	Original formats	Wiki-viewable formats
Project schedule (A)	Gantt chart (MS Project)	Image (jpeg)
Developer feedback (A)	Email message	Wiki text
Research data (A)	Web content (BackpackIt)	Wiki text
Project schedule (B)	Gantt chart (MS Project)	Table (HTML)
User requests (B)	Spreadsheet (MS Excel)	Table and text
Business roadmap (E)	Word document	Wiki text

Formatting wiki content. Formatting content was mentioned by many participants and evidenced in shared documents as a way to support transferring content. The functional requirements document in Project B and the use case document in Project C were examples of how the design teams used formatting to help the project members focus attention on important information. In other cases, participants explicitly stated that they formatted the content so that others could read it more easily. For example, the Project A manager revised content format of a meeting agenda so that the team could read it easily.

In general, the project teams employed basic formatting because advanced formatting was too difficult. Simple wiki markup syntax, such as bold text (''' '''), italic text ('' ''), section (=== ===), bullet list (*), and numbered list (#), was frequently used throughout the project wiki sites whereas HTML tags, such as table (<table><tr><td>… </td></tr></table>) and preformatted text (<pre style= “”>… </pre>, were seldom used. There were only 4 tables created in the Project A wiki site and 14 tables in the Project B wiki site. Commonly used formats are sections and headings, bold, and bulleted lists as shown in Table 6-7 below.

Table 6-7: Wiki formats

Formats	Project A (76 pages)	Project B (46 pages)
Section	51	24
Bold	24	20
Bullet list	13	18
Numbered list	3	14

Sixteen participants reported that the wiki did not support complex editing such as those provided in Microsoft Word. Thus, some participants would prefer to create a document in Word rather than in the wiki.

> *"The wiki table is a bit cumbersome to edit, I just told people look for the word 'facilitator' [when editing the usability study schedule]." – A:Designer 1*

> *"The standard wiki formatting is very linear so if people want to have different arrangement of the information you have to take the time to make html table and to figure out how it's looked on the page." – B:Project manager*

> *"The one thing that I think is our lessons learned – we will not use it again to document and edit use cases on the wiki itself. We'll put them in a different document and link them in. Adding is just a nightmare. We wasted a lot of time trying to get them formatted right when it'd just be easier to put them in Word and then just link that document. It's quick to edit it, but the one thing that's not quick to edit is something like use cases where you need nice formatting." – C:Project manager*

Displaying viewable-size images. Most images were created and shared by designers. Due to the flexibility offered by the MediaWiki, designers could choose what size these images should be displayed in, ranging from default small thumbnails (Figure 6-5) to actual-size image.

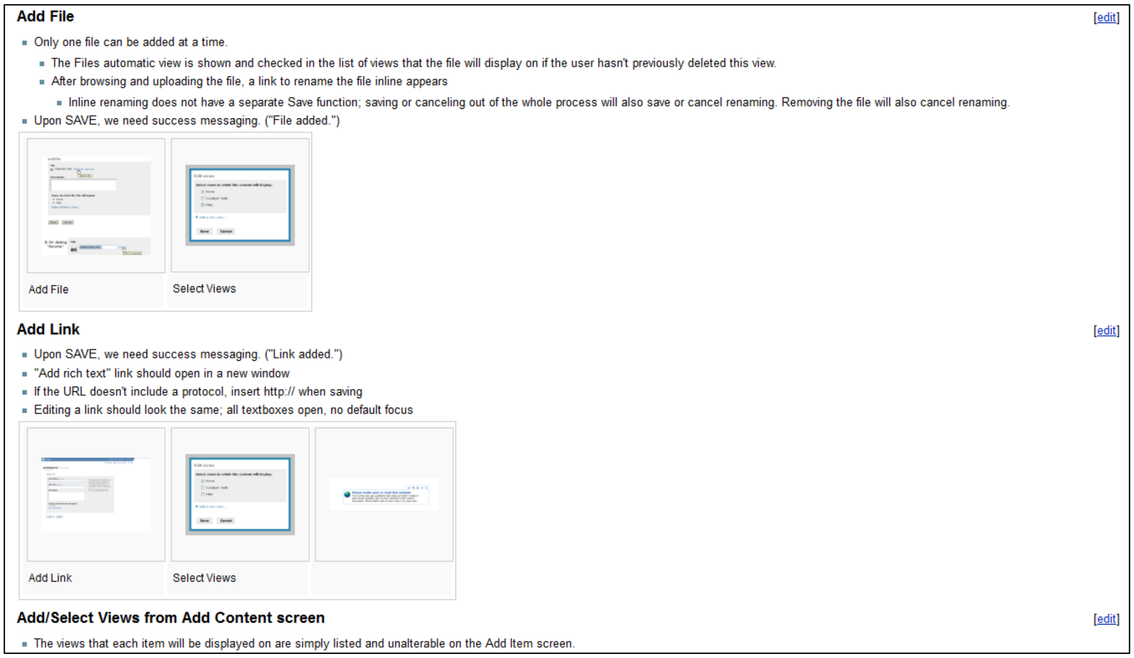


Figure 6-5: Images displayed as thumbnails

There was no standard or guideline on how to post images on the project wiki sites. The designers used their own judgment on context and purpose of shared images to decide on an image's display size. By making images viewable on a wiki page they were posted, for example, B:Designer1 chose a large display size for screenshots in the design specification as shown in Figure 6-6 because he would like all images to be viewable right on the wiki page as he explained, *"half the time people would just wanna see the picture of how it looks. And so what that would mean for them is they gotta click on the thumbnail because the thumbnail's not that useful. So this way it sorta flipped it around, made the picture bigger. Hopefully, it describes itself for the most part."* In fact, the designer's decision to make a large display size for screenshots worked well for the developers as one developer said that *"Most of the time, I click through all the images, and I actually go in and blow them up just so I can see them in good detail here."*

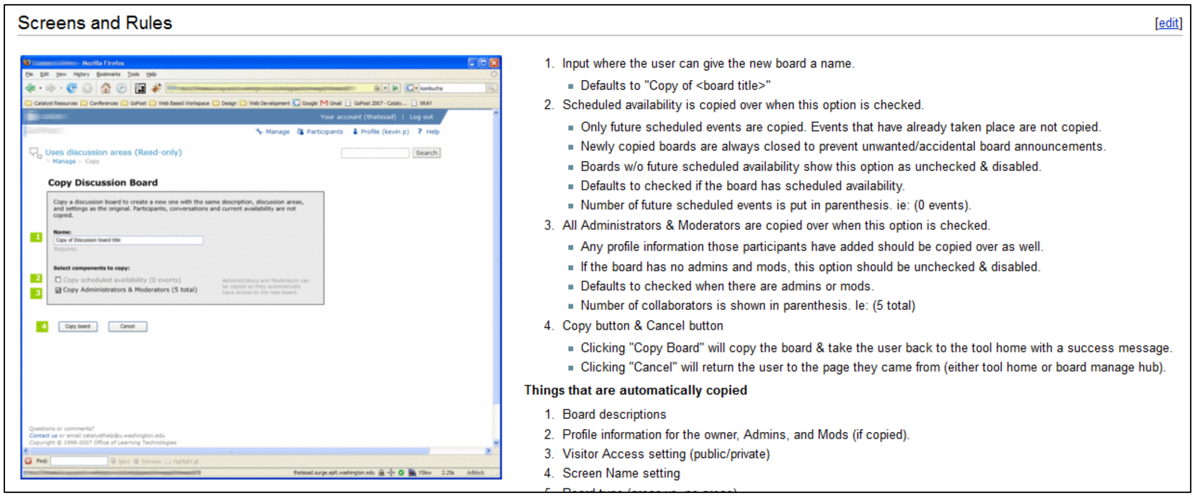


Figure 6-6: A viewable-size image

6.4.3 Semantic sharing: Translate

Once the shared information artifacts were viewed, the design teams engaged in the next level of sharing: semantic sharing. At this level of sharing, the design teams aimed at sharing the interpretation or meaning to ensure that project members from different disciplines had the same understanding of the shared information. Design project members from different disciplines could interpret the same information differently as described by one designer, "*Because a lot of time just because [of] different teams it's hard to. I mean we just don't speak the same language: different backgrounds, different ways of understanding, the concerns at any given point.*" When a shared understanding was needed for artifacts shared on the project wiki sites, the information sharers took actions to communicate and confirm the meaning by providing clarification or explanation, having a discussion, and revising content.

Providing clarification and explanation. After making sure that the content is readable, and actually read by team members, participants wanted to make sure that their understanding was also shared as A:Designer2 described, *"You have to constantly with anything you put up there, you have to put it in a way that would be understood by the team."* It is particularly important for interdisciplinary collaboration, in which people had different perspectives. He also explained that designers tried to include design rationales in both meeting notes and design spec because they *"want to make sure that it's clear"* to the developers. Similarly, the E:Development manager would like to have the terminology defined and available on the wiki as he described, *"Those terms are defined in the wiki and used in the tests. So I want the terms to be defined in the wiki so that everyone can understand."* Participants often clarified the wiki content outside of the wiki, mostly face-to-face as I observed in Project A and Project B that project team members presented and explained the wiki content during a team meeting.

To make the meaning clear to others through the wiki, the design teams added descriptions or explanations close to the content on a wiki page. Clarification or explanation on the wiki page often had a different formatting treatment than content, and was added to with a purpose of clarifying the content rather than generating discussion. For example, the designers in Project B added their design rationale next to or below screenshots in the design specification to help the developers understand their design decisions (Figure 6-7 and Figure 6-8).

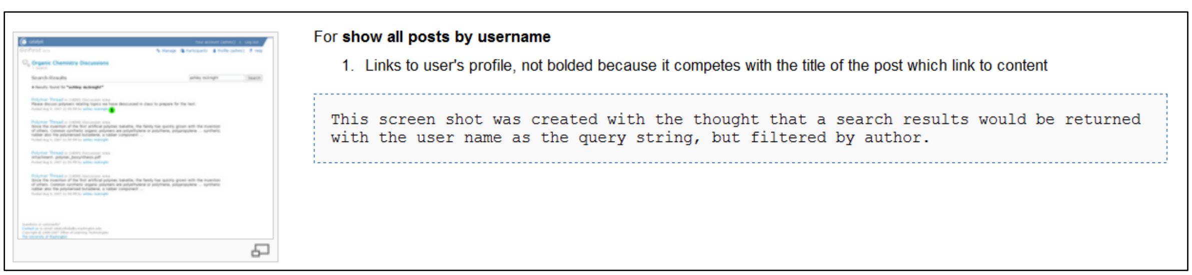


Figure 6-7: Clarification and explanation on the project wiki sites

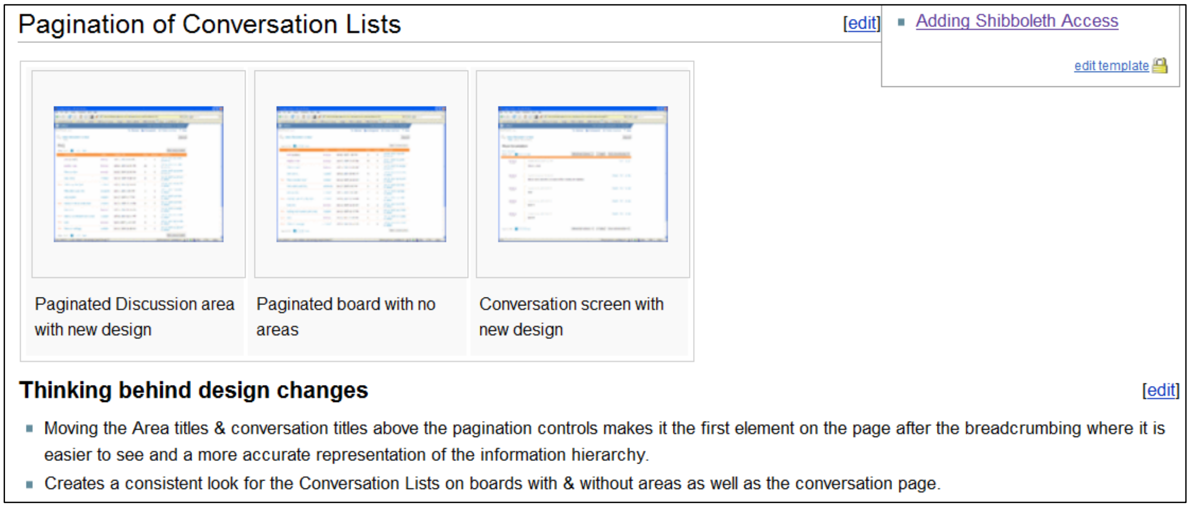
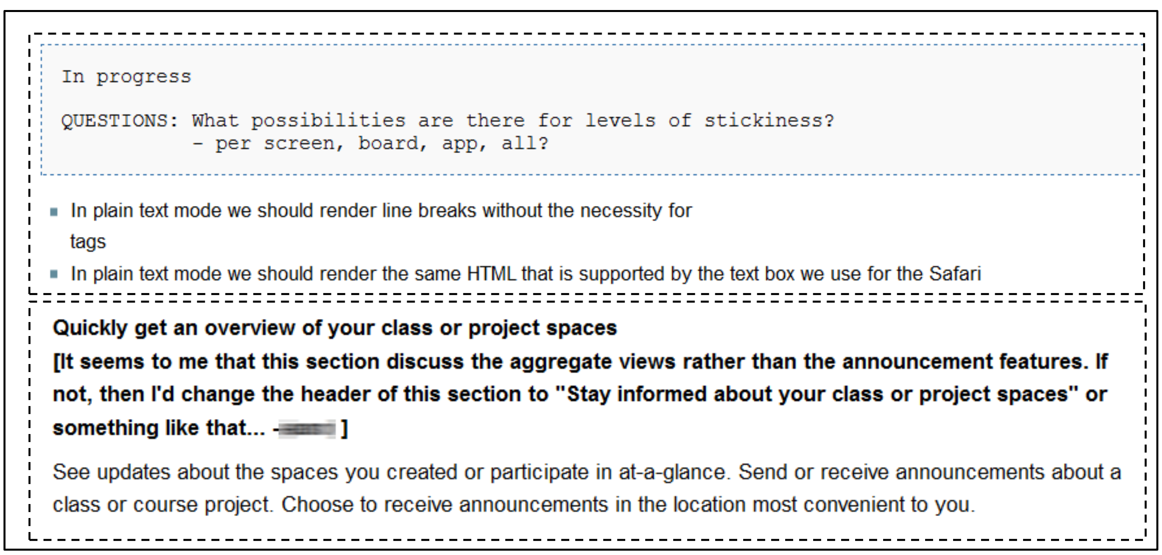


Figure 6-8: Providing design rationale for explanation

Having a discussion about artifacts shared on the project wiki sites. Most wiki platforms provide a discussion space for wiki editors to leave notes about or discuss changes to its associated wiki page. MediaWiki, used at State University and Digital Media, has a corresponding discussion page easily accessible on the top of every wiki page. OpenWiki, used at Flynow, however, does not provide a discussion space. The project teams that used MediaWiki (Project Team A, Project Team B, and Project Team E) did not use project wiki sites to hold their discussions. The project members occasionally posted a question or comments on the project wiki site. However, in Project A and Project B, very few posted questions, and comments were responded in the project wiki sites. Through my observations, these posted questions or comments were used to trigger a discussion that happened outside of the wiki as in a meeting, email, or chat. One designer from Project B expressed his interest in using the wiki as a channel for having a discussion asynchronously as he said, *"So if…say a developer is looking at this spec. I don't – he might not want to – as soon as he has a question, come and ask that question right away. Like, sometimes it just works a little better asynchronous. So if somebody* *could just say, oh, what does this mean here? Like write something on it, right. And then I could come in maybe the next day or whatever and tackle those questions and answer those."* Similarly, the project manager of Project D described her previous use of wikis to communicate with a developer as she explained, *"There was a project I was on a couple of years ago where I worked with a particular developer. She was just asking me lots of questions about an area that I had expertise in about an airline system. We used the wiki, and she had a link to a wiki page, and she would just put questions on there. As I answered them, I would put an emoticon next to it showing that this one's done. It was a long list. There'd be a smiley face or there might be something to indicate oh, I see this, but I don't know and I'm looking into it."*

Unlike Wikipedia contributors, these co-located software design teams did not use the discussion page, the designated place for discussion about a content page provided by most wiki platforms. Instead, the project teams posted questions on a wiki page to initiate a discussion as illustrated in Figure 6-9.



In progress

QUESTIONS: What possibilities are there for levels of stickiness?
- per screen, board, app, all?

- In plain text mode we should render line breaks without the necessity for tags
- In plain text mode we should render the same HTML that is supported by the text box we use for the Safari

Quickly get an overview of your class or project spaces

[It seems to me that this section discuss the aggregate views rather than the announcement features. If not, then I'd change the header of this section to "Stay informed about your class or project spaces" or something like that... -[redacted]]

See updates about the spaces you created or participate in at-a-glance. Send or receive announcements about a class or course project. Choose to receive announcements in the location most convenient to you.

Figure 6-9: Evidence of discussions on the project wiki sites

On the MediaWiki platform, used at State University and Digital Media, a discussion page was available as a tab on all wiki pages (see Figure 6-10). The discussion tab label is red for all pages of the two project wiki sites. The red color indicates that the page has not been created. Participants found the discussion page too separated from the content, where they would like to have a discussion.

> *"I think the closest thing they have is, like, the discussion page, which is really just another wiki page. And it's very hard to, like, relate content to the discussion. Like, nowhere do you get to see both at the same time, so you have to go back and forth - that's just slow. I don't think anybody wants to do that." – B:Designer 4*

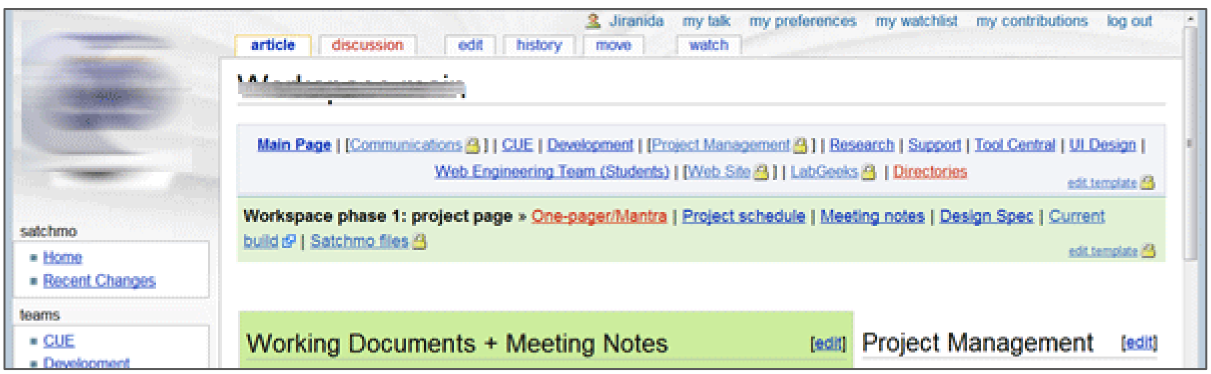


Figure 6-10: Discussion tab on a wiki page

The design teams preferred having more direct communication, during which the shared artifacts would be displayed on a big display as the team discussed, asked questions, and gave comments. This process is to ensure that the project members had the same interpretation of the information. Although the discussion did not take place in the project wiki sites, the evidence of discussions was observed in the wikis. The most obvious evidence was the record of discussion and its results in the meeting notes page.

"I'm a big face-to-face person so I get up and I go talk to [the designers]." – A:Researcher1

"It's like, wait a minute, I'm sure the spec didn't used to say that. What changed? You can see who changed it, and when they changed it, which I find nice. You can go talk to a person, and ask why are you making this change?" – B:Developer2

"There was other stuff on there, too, that was a little bit beyond my understanding that - I'm sure it works for them. […] And the only things that I didn't like is it got into too technical of detail for what I do because that's the point where I would have to call somebody and say, 'What in the world does it mean?' But I'm sure for people who are using that a little more literally, it'd be quite matter-of-fact - but, at least, it's for me" – D: Customer care2

Revising content shared on a project wiki site. There are two types of revisions: major revision and minor revision. Major revisions, often a result from a discussion, or a team review, and create a new understanding of shared artifacts. Major revisions observed on the project wiki sites are: adding or removing content; rewriting a substantial amount of content; changing images, uploading a new version of a file; and linking to a new version of artifacts. For some information artifacts such as functional requirements documents, screen flow diagrams, and design specifications, the review-and-revise loop occurred multiple times until the whole team agreed on a shared understanding, which led to multiple revisions of artifacts as observed on the project wiki sites.

On the other hand, the project teams also made minor revisions to artifacts shared on the project wiki sites. Minor revisions were editorial revisions that did not change the meaning of existing text such as line-spacing, spelling, grammar, and word choice. Figure 6-11 shows examples of editorial revisions by comparing difference between versions on the History tab. The editorial revision was observed more frequently on the wiki pages containing design artifacts and framing artifacts (e.g., use cases, functional requirements, and design specifications) but not the wiki pages containing procedural artifacts (schedule and meeting notes).

-	Easily grant access and control availability	+	Easily grant access and control availability **of information**
-	Easily **provide access to the spaces to the** students **in a class you are teaching,** or members of a work group. Hide information while you are editing it; when ready, make it available to class or group members.	+	Easily **grant** students or members of a work group **access to a space you created**. Hide information while you are editing it; when ready, make it available to class or group members.
	Share the workload with colleagues		Share the workload with colleagues
-	Give collaborators access to help you with the class or project **context**. Transfer **a context** to another instructor or account.	+	Give collaborators access to help you with the class or project **space**. Transfer **an online space** to another instructor or account.
	Prepare for next quarter		Prepare for next quarter
-	Archive the materials you need online or **in** download to your local computer. Copy an entire space (Catalyst tools not included), and then modify it to get ready for the next class you are teaching or project you are working on.	+	Archive the materials you need online or download to your local computer. Copy an entire space (Catalyst tools not included), and then modify it to get ready for the next class you are teaching or project you are working on.

Figure 6-11: Wiki content revisions

Revising wiki content created by other project members resulted in a co-authoring activity. The wiki made it easier to co-author a document. Sometimes each project member took a different section while other times they revised existing content. The co-authoring activity was observed in Project A and Project B as shown in Figure 6-12. The project teams, particularly Project B, used wikis as a collaborative authoring tool. The wiki log analysis reveals that many wiki pages were co-authored (85% for Project A and 49% for Project B).

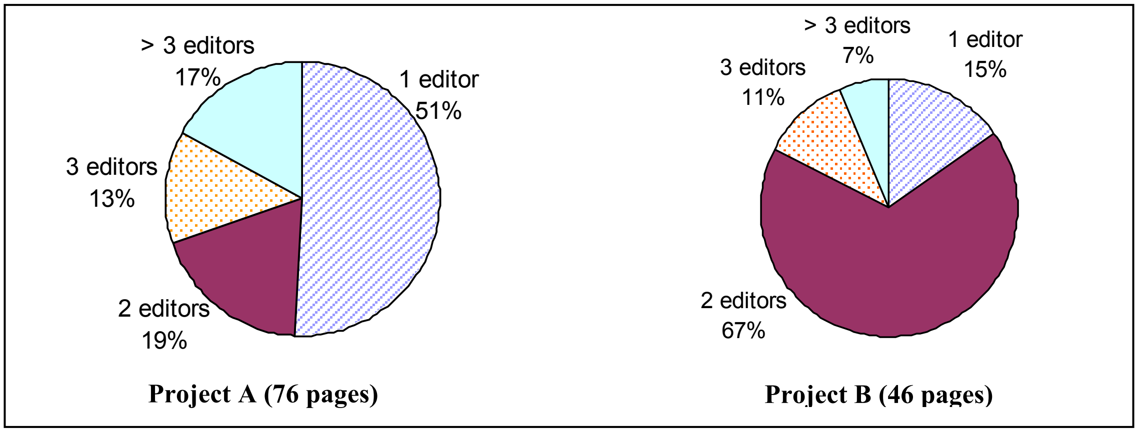


Figure 6-12: Distribution edits by number of editors per page

The wiki pages of Project A with a high number of editors were the meeting notes page (10 editors), the project wiki homepage (7 editors), the project adoption ideas page (6 editors), the usability study pages (5 editors), the literature review summaries page (4 editors), and the workspace scenarios page (4 editors). The project pages of Project B with high number of editors were the project schedule page (7 editors), the meeting notes page (5 editors), and the feature list from previous project page (4 editors). These wiki pages indicate design activities with high collaboration.

I further analyzed the wiki pages with 2 or more editors to reveal interdisciplinary collaboration through the wiki content editing. The findings show that 24% of Project A's wiki pages (n=76 pages) and 35% of Project B's wiki pages (n=46 pages) were edited by members from multiple disciplinary groups. The designers contributed to all but one co-authored pages from both project wiki sites.

6.4.4 Pragmatic sharing: Transform

For pragmatic sharing, the design teams were interested in the use of shared artifacts – how project members from other disciplines could easily use the information to accomplish their tasks. The sharing process at this level is called *transform*. The project teams used the project wiki sites to support transformation by organizing content, providing status information, and creating and utilizing shared task lists.

Organizing content: To transform information artifacts, design teams structured the content in order to make it easy for intended audience to use the artifacts. This process requires an understanding of intended use. For example, a project schedule page was reorganized in order to make it easy to update work progress, and the design specifications pages were organized to better served developers' needs when using the artifact during their coding.

> *"I broke the huge table into separate tables for each phase just to make editing it later a little easier. Feel free to revert it back using the wiki history if this ends up being a pain." – B:Designer1*

> *"It's organized quite a bit better. The design specification it's broken out into the three different phases. Right now, we're working in phase two. Everything's been broken out by large categories of the work. [….] I just kind of ignore everything else and go to that. I just scroll through here and look – usually I go for the screen shots first." – B:Developer2*

The design teams created and employed a wiki document template on multiple wiki pages that contain the same type of content, for example, usability study notes, the usability study schedule, and test cases. A wiki document template was particularly useful when multiple project members were involved in creating documents. When the designers and researchers from Project A were conducting a usability study, one designer developed a wiki document template for usability study notes by creating wiki pages for all study sessions. Those wiki pages contained the same section headings as illustrated in Figure 6-13. With the template, the usability study notes could be analyzed more efficiently because data was recorded in the same way.

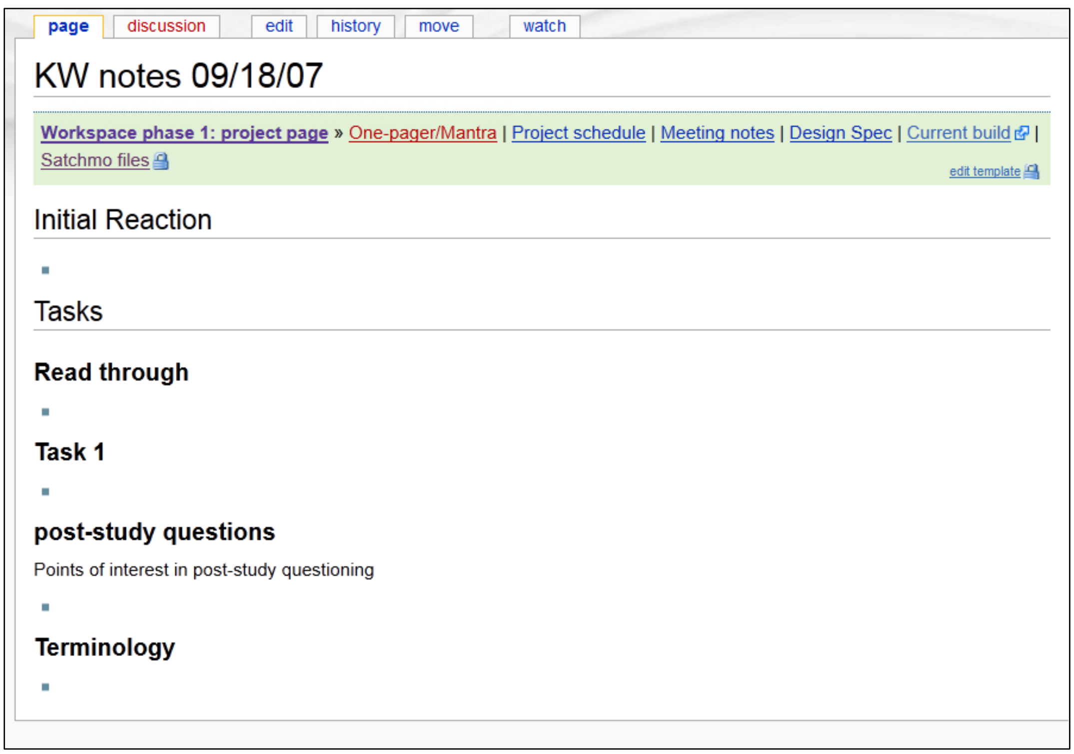


Figure 6-13: A Wiki document template for usability study notes

Providing status information: Two project teams (A and B) provided status information for information artifacts shared on the project wiki sites in order to support their hand-off process. A hand-off process occurred when a shared artifact was created by one disciplinary group, but was used by a different disciplinary group. Before the wiki adoption, participants at EdApp, State University, reported that artifacts were not accessible to project teams until they were finished. Thus there was a clear hand-off. With the project wiki sites, however, artifacts that were created as wiki content were always available to all project members. To accommodate the hand-off process in project wiki sites, the teams added status information to the artifacts to let the intended users know whether the artifacts were ready for use. Status information was added as a link label or in the wiki content itself. For example, the designers from project B posted status information on each design specifications wiki page as shown in Figure 6-14.

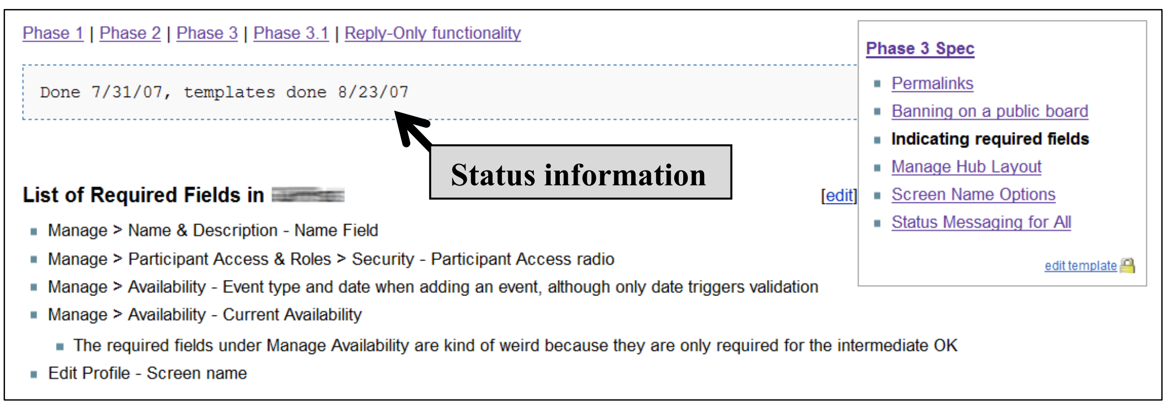


Figure 6-14: Status information for the design specifications on a project wiki site

Both designers and developers agreed that having status information on a wiki page was helpful as they explained:

> *"So what we've done at the top of those spec pages, there's is like a little bar that says in progress, don't work on this right now." – B:Designer1*

"It's more prone to have status information on it. You'll see the pages where it'll tell you that things are not ready for development use, and are marked as that. Unlike before, when I would just have a big pile of graphics to look at. Now, there's some sense of where things are in the process." – B:Developer2

Another way to provide status information was through naming and labeling. Design teams labeled wiki links to reflect the document version or status such as "Charter V.3 - Approved & Update on 3/6", "Scope By Subject Area – Refined", and "Scope Brainstorm List - Initial list". Using descriptive link labels, B:Project manager chose to use a combination of link labels and headings to communicate the status of functional requirements as shown in Figure 6-15. With these descriptive link labels, the designers could refer to the right version to guide their design work.

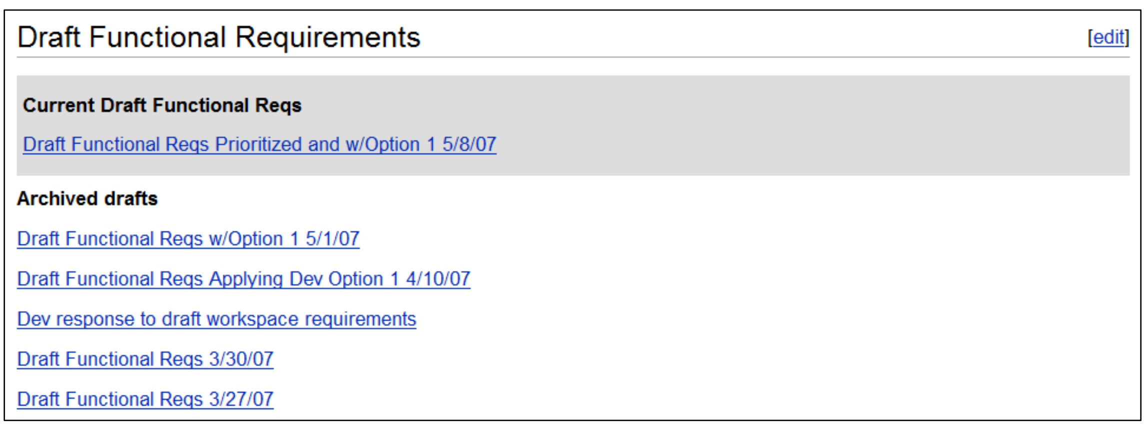


Figure 6-15: Status information through descriptive naming and labeling

Creating and utilizing shared task lists: Three project teams created task lists on their project wiki sites to track cross-discipline work. Participants used the shared task lists to identify what needed to be done and what they could work on. One designer explained that he created a wiki page for tracking front-end development work that he collaborated with developers – *"to sort of prioritize and just make lists, like a pretty comprehensive list of what needed changing. It was a shared task list. We'd strike things through when we were done. And so I could see what he had worked on and I knew what things that I could work on if I needed to."* Table 6-8 shows a list of shared task lists created and shared on the project wiki sites.

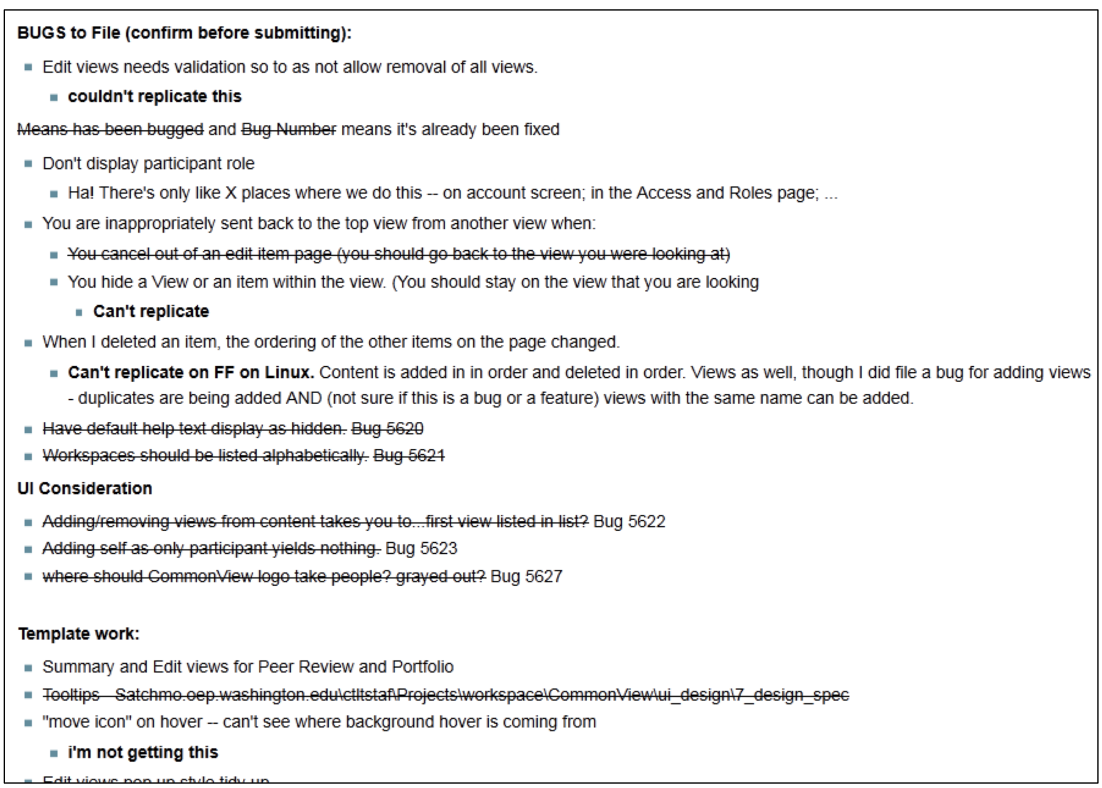
Table 6-8: Example Shared task list wiki pages

Wiki document	Purpose	Task list users
[A] Template Tidy Up	A shared to-do list to support collaboration between designers and developers	Designers Developers
[A] Needed functionality	To keep track of what functionality the developers needed to develop before the application could be release	Designers Developers
[A] Workout	To keep track of design issues that needed to be resolved as a team	Project manager Designers Developers
[A and B] Next Push Checklist	To keep track of what needed to be done on the release date	Developers Testing QA
[B] Schedule	To track work progress of the project	All
[E] Prioritized task list	To track what tasks needed to be done in a current sprint	Project manager Scrum master Developers QA

The development group at EdApp also had a wiki page they used to collaborate with the QA group on what needed to be done on a release date. It was called the "Next Push Checklist" as a QA engineer described, *"People can edit it all at once, so all the developers and I can all be editing the wiki, marking off what we're doing, as we're doing it, while we're releasing."* The

QA engineer reported that this checklist had made the releases go a lot smoother, because they could make sure everything actually got done on the release date. Moreover, the checklist had reduced the down time (*"because everyone could work at once")* and the amount of error of the release process. The Next Push Checklist wiki page was not part of the project wiki sites, but was linked from both the QA wiki page and the developers' wiki page.

The project members who were involved in the tasks listed in a shared task list added subtasks that need to be done by others and checked off tasks that they had completed. Participants had different ways to represent to-do tasks and completed tasks. The data from Project A and Project B show that completed tasks were either marked with completed progress, crossed off (strike-through), or removed entirely from a wiki page as shown in Figure 6-16.



BUGS to File (confirm before submitting):

- Edit views needs validation so to as not allow removal of all views.
  - **couldn't replicate this**

~~Means has been bugged~~ and ~~Bug Number~~ means it's already been fixed

- Don't display participant role
  - Ha! There's only like X places where we do this -- on account screen; in the Access and Roles page; ...
- You are inappropriately sent back to the top view from another view when:
  - ~~You cancel out of an edit item page (you should go back to the view you were looking at)~~
  - You hide a View or an item within the view. (You should stay on the view that you are looking
    - **Can't replicate**
- When I deleted an item, the ordering of the other items on the page changed.
  - **Can't replicate on FF on Linux.** Content is added in in order and deleted in order. Views as well, though I did file a bug for adding views - duplicates are being added AND (not sure if this is a bug or a feature) views with the same name can be added.
- ~~Have default help text display as hidden. Bug 5620~~
- ~~Workspaces should be listed alphabetically. Bug 5621~~

UI Consideration

- ~~Adding/removing views from content takes you to...first view listed in list?~~ Bug 5622
- ~~Adding self as only participant yields nothing.~~ Bug 5623
- ~~where should CommonView logo take people? grayed out?~~ Bug 5627

Template work:

- Summary and Edit views for Peer Review and Portfolio
- ~~Tooltips - Satchmo.oep.washington.edu\ctltstaf\Projects\workspace\CommonView\ui_design\7_design_spec~~
- "move icon" on hover -- can't see where background hover is coming from
  - **i'm not getting this**

Figure 6-16: Project A's shared task list wiki page

For Project E, the project manager chose to annotate tasks with color-coded annotations – "Complete" in green and "In Progress" in yellow as shown in Figure 6-17. Thus, these shared task lists were dynamic and short-lived. Once the tasks were completed, these wiki pages were useless; they did not contain any information needed in the future or were empty.

Customer Web Facing Team

PM : [Name] | **Status : Green** | **Date :** 23-Mar-2009

[edit] **Chat on Search Results**

Objective: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras dictum erat dictum leo tincidunt id suscipit neque scelerisque. Vivamus aliquet tristique turpis eget euismod.

Next Release to Production: TBD

Items Included in Release:

- Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras dictum erat dictum leo tincidunt id suscipit **Complete**
- Proin enim arcu, eleifend sed convallis vitae, iaculis congue enim. **Complete**
- Donec ultrices elit ac nibh dapibus lacinia mauris rutrum. Proin euismod dapibus **In Progress**

Upcoming Milestones/events:

Issues/Risks::

- Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras dictum erat dictum leo tincidunt id suscipit

Figure 6-17: Project E's shared task list wiki page

A few participants, however, stated that they did not like using wikis for a shared task list because the wikis did not provide synchronous updates. Participants needed to refresh the page to see changes. Therefore, some participants used another tool such as a Google Docs and

Gobby. B:Developer1 used Gobby (an open source, collaborative text-editing tool) to share a to-do list with B:Designer1 during the development and template tasks. He explained that they chose to use Gobby rather than the project wiki to create their to-do list because *"it's live, unlike [the wiki]. It makes it easy to communicate up to date information."*

6.5 Summary

This chapter presented the findings on the ways in which software design teams used wiki technology to support information sharing across disciplines, including the reasons why the design teams adopted wiki technology, information artifacts shared on the project wiki sites, information sharing needs served by the project wiki sites, and how the design teams actually shared information artifacts on the project wiki sites. The findings revealed that the software design teams used the wiki technology as their central information-sharing tool because it provided a mechanism to link heterogeneous information from various sources into one place. The information artifacts shared on the project wiki sites could be categorized into three types based on their use purpose: framing artifacts, design artifacts, and procedural artifacts. The findings revealed that the software design teams used wiki to meet their information sharing needs, which were to provide access to project-related information, to keep project members up to date, to build a shared understanding, and to complete a task.

Additionally, the findings provided empirical evidence that the design teams engaged in multiple levels of sharing when they shared information across disciplines using the wiki technology. Using the 4-T framework, individual information sharing processes were further discussed, along with issues the design teams faced, as summarized in Table 6-9.

Table 6-9: Information sharing on project wiki sites summary

Levels	Processes	Wiki features	Issues
Transport	Creating a wiki site Adding information artifacts Linking to shared information artifacts Organizing the project wiki site Pushing information artifacts	Linking Category (tagging) Recent changes	Finding information Orphaned pages Project wiki page template
Transfer	Converting content to wiki-viewable format Formatting content Displaying viewable-size images	Wiki markup syntax	Advanced formatting
Translate	Providing clarification and explanation Having a discussion Revising content	Talk page Revision history (versioning)	Discussion is not integrated with content.
Transform	Organizing content Providing status information Creating and utilizing shared task lists	Wiki page template	No real-time update

CHAPTER 7: DISCUSSION AND CONCLUSION

This chapter summarizes the findings, discusses the key findings, presents the practical implications and the theoretical contributions, and concludes with limitations of the study and suggestions for future research.

7.1 Summary of Findings

This section summarizes findings, organized around the three research questions: (1) How did the design projects share design-relevant information across disciplines?; (2) To what extent do wikis support information sharing and interdisciplinary design activities?; and (3) How could wikis be enhanced or utilized to better support information sharing in interdisciplinary design?

RQ1: How did the design projects share design-relevant information across disciplines?

With its simplicity and flexibility, the wiki technology has provided a great paradigm for investigating information sharing across disciplines in dynamic environments such as software design projects. The wiki technology enables flexibility in information sharing because it did not limit software design teams to any particular way of sharing. Thus, the findings uncovered the conceptual issues of information sharing including information collection, information-sharing needs, and information-sharing processes.

Software design projects create and share various types of information, which are heterogeneous in nature and purposes (Aguiar & David, 2005; Davis et al., 1999; deSouza, 2003; Walz et al., 1993). Different terms have been used to refer to information items shared during a design process, for example, documents (Poltrock et al., 2003, Yitzhaki & Hammershlag, 2004), coordinative artifacts (Schmidt & Wagner, 2004), and design representations (Hendry, 2004). In this dissertation, I proposed a new typology of information items shared among disciplines. These three types of shared information items are framing artifacts, design artifacts, and procedural artifacts. This typology represents a relationship between information sharing and three components of a design situation, which are design problem, design process, and design solution. The framing artifacts were shared to build a shared understanding of a design problem that the software design team was trying to solve. The design artifacts represented a proposed design solution that the software design team was working on. The design artifacts were shared to gather feedback and different perspectives from various disciplinary groups. The procedural artifacts were shared to support design processes and tasks, which were shared among disciplines during tasks that require collaboration across disciplines (e.g., bug fixing, software release).

Similar to previous software design research, the findings reveal that software design teams shared information across disciplines to get their tasks done and to keep the project team well informed about the project. Furthermore, I identified four information-sharing needs that motivate software design teams to share information. These information-sharing needs are: (1) to provide access to project-related information; (2) to keep project members up to date; (3) to build a shared understanding; and (4) to complete a task.

To meet the information-sharing needs, the design teams engaged in multiple levels of sharing when they shared information across disciplines. With the 4-T framework, I was able to examine the information sharing processes that occurred across disciplines during the software design process. The findings reveal that software design teams engaged in transporting, transferring, translating, and transforming processes during their information-sharing activities. Figure 7-1 illustrates four levels of information sharing and associated processes. As the sharing moves up, from the bottom (materialistic sharing) to the top (pragmatic sharing), the complexity of sharing and collaboration among disciplines increase. The area within a triangle represents the relative number of shared artifacts at each level. In other words, there are more artifacts being transported across disciplines than that being transferred, translated, or transformed.

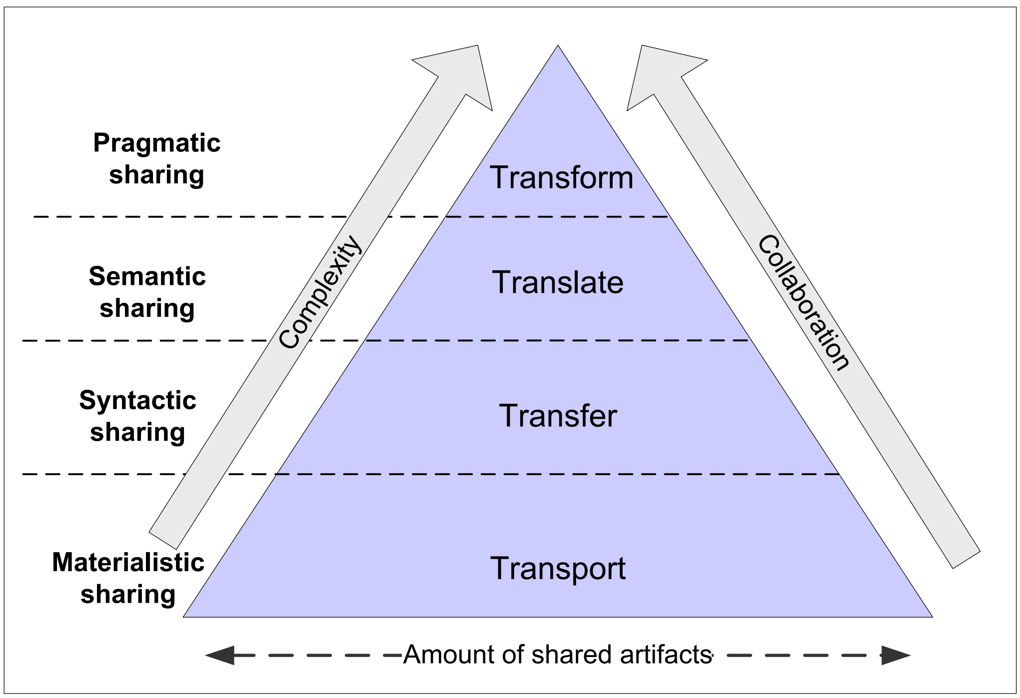


Figure 7-1: Information sharing from the 4-T lens

RQ2: To what extent do wikis support information sharing and interdisciplinary design activities?

It has been reported that the heterogeneous nature of information items and sources (i.e., different formats, different systems) demanded software design teams to employ various information sharing tools (Aguiar & David, 2005). Software design teams used wiki technology to provide easy access to project information. The project wiki site did not replace any existing information sharing tools. Rather, it served as the glue that linked heterogeneous information from various sources into one place. The wiki technology made it easy for software design teams to aggregate information items from different tools. Thus, a project wiki site can be considered a common information space (Bossen, 2002; Carstensen, 2000; Davis et al., 2001), within which software design teams share all three types of information (framing, design, and procedural artifacts).

Using the 4-T framework, along with the CWA, to assess the usefulness of wiki technology in supporting information sharing across disciplines, I found that: (1) Wikis were used primarily to support transporting and transferring, that is, giving access to and delivering content of information items; (2) Wikis had the potential to support translating and transforming, that is, discussing and handing off information items, but teams appeared to have difficulty utilizing features that would enable these forms of information sharing; (3) Software design teams did not fully utilize the wiki's key features such as adding category tags to wiki pages, which make it easy to browse related wiki documents, and using a discussion page associated with each individual wiki page to record related discussion; and (4) The use of wikis appeared to be influenced by many social and technical factors, including organizational environments, the nature of the design project, the nature of collaboration, the design tasks, shared artifacts, project team members' experiences and attitudes, and the specific kinds of wiki platforms that were used.

The findings also revealed five challenges of project wiki sites. First, there was a challenge balancing between the structure and flexibility of project wiki sites. By structure I mean defining structure of a project wiki site before creating content, and by flexibility I mean allowing the organization of a project wiki site to grow organically as content being created. As discussed in Chapter 6, most project teams valued the wiki page templates because a predefined structure provided guidelines on where to put information as well as a consistency across projects. However, too much structure can diminish the usefulness of project wiki sites, as experienced by Project Team A. Grudin and Poole (2010) discovered a similar challenge in the content organization of team wiki sites when they examined wiki use in enterprise settings. They reported that a wiki site structure did not evolve to meet the evolving needs of wiki users.

Second, finding specific information on the project wiki sites can be difficult due to a large amount of content. This issue could easily be addressed with a decent search engine; however, most wiki platforms offer only a title search, while some have full-text search. For the wiki platforms investigated in this dissertation, their search function was recognized as an issue by all five project teams. The project teams reported that the wiki search did not return useful results. As a result, the project teams had to use other information-sharing tools to ensure that the team members could find specific artifacts. For example, team members still relied on instant messenger or email applications to notify others that the information had been shared on the project wiki sites with links to specific wiki pages in their notifications.

Third, the software project team members did not contribute equally to project wiki sites because the wiki technology currently employed by their team did not support all tasks. Although the wiki adoption was initiated by software developers, project wiki sites were contributed to by project members from other disciplines (especially project managers and designers) more than by software developers. For example, at EdApp (State University), designers used the project wiki sites as their working space where they created and recorded most of the design artifacts, including task flow diagrams, usability study information, and design specifications, while software developers and QA engineers used the project wiki sites mainly to receive information from designers. Software developers accessed project wiki sites often during the development phase as they looked at design specifications to guide their coding. However, they did not share information through the project wiki. As a result, designers contributed to the project wiki sites more than other disciplines. Software developers, on the other hand, who were the first to use the wikis and introduced them to the whole group, did not contribute to the project wiki sites as much as other disciplines did.

Fourth, the project team members did not take a full advantage of the open-editing nature of wiki technology; they rarely edited each other's content. Although the project team members had a desire to revise or reorganize existing wiki content, they did not feel comfortable making substantial edits on what they called 'other people's content.' The majority of contributions observed in the study were adding content or making editorial revisions (e.g., typo fixing and grammatical corrections). A similar behavior has been reported in the Wikipedia co-authoring environment, in which a sense of ownership and territorial behavior has discouraged new contributors to participate (Thom-Santelli, 2009), and in the corporate wiki environment, in which people are reluctant to modify others' content (Danis & Singer, 2008). Rader (2009) also discovered that users of a group filing system were reluctant to modify or delete files that they did not own.

Finally, the majority of project wiki contributors were either minimalists or adders, as defined by Majchrzak et al. (2006). Based on the survey results of 168 corporate wiki users, Majchrzak et al. (2006) identified four types of contributors based on the amount and characteristics of their contribution. The wiki contribution can be measured as low or high, and characterized as adding or synthesizing. Adding contributions involve adding content and adding pages, while synthesizing contributions involve integration, reorganization, and rewriting content. The four contributor types are minimalists (low adding, low synthesizing), adders (high adding, low synthesizing), synthesizers (low adding, high synthesizing), and multiplexers (high adding, high synthesizing). Minimalists are project members who add little new content and hardly revise existing content on a project wiki. Adders are project members who add a lot of new content, but hardly revise existing content on a project wiki. Synthesizers are project members who add little new content, but make many revisions to existing content on a project wiki. Multiplexers are project members who make the most contributions to a project wiki. They add a lot of new content and make many revisions to existing content on a project wiki. For all five projects, most project members were identified as either minimalists or adders whereas there were only a few synthesizers and very few multiplexers. In Project A and Project B, there were three multiplexers, all of whom were designers who had a lot of wiki experience prior to using the project wiki sites.

RQ3: How could wikis be enhanced or utilized to better support information sharing in interdisciplinary design?

The answer to this research question is presented in the form of wiki technology requirements in order to better support information sharing across disciplines during software design process.

The requirements are informed by the context dimensions (previously discussed in Chapter 5) that influence the use of wikis. The requirements are summarized in Table 7-1.

Table 7-1: Requirements for wiki technology

Context dimensions	Requirements
Work environment: Large organizations	• Scalable • Interoperable among wiki platforms
Work domain: Web application development Projects	• Support agile web-application design and development activities • Integrate with common tools used in software design • Provide a granular access control to restrict access at multiple levels (e.g., at a wiki site level, at a page level, and at individual user level)
Social organization: Interdisciplinary team	• Foster cross-disciplinary collaboration among project members • Support negotiation across disciplines
Activity: Information sharing through project wiki sites	• Support sharing of various types of artifacts • Provide access to all project information • Support all levels of sharing (materialistic sharing, syntactic sharing, semantic sharing, and pragmatic sharing)
Actor's resources and value: Team member profiles and attitudes	• Easy to learn • Easy to use • Flexible in content organization • Open editing

Requirements informed by the work environment include: scalable, interoperable among wiki platforms, and supportive of an agile development process. The adoption of wiki technology began as grassroots effort in the IT departments of the three organizations studied in this dissertation, which could be common at other large organizations. That is, IT department staff set up a wiki site for their departmental internal use, and then the wiki became widely used throughout the organization. Therefore, the scalability of wiki technology is important. The wiki technology should be able to handle a growing amount of content and users ranging from a few users in one organizational unit to thousands of users in an entire organization. The wiki technology should be interoperable among various wiki platforms because the grassroots adoption may result in an implementation of multiple wiki platforms. At Digital Media, for example, different organizational groups started their own wiki sites on different wiki templates, which were MediaWiki, Blogit, and MonMonWiki. In order to consolidate content from different wiki platforms, organizational wikis need to be able to work with existing wiki platforms. At a minimum, the organization should be able to migrate content from multiple wiki platforms to the organizational wiki platform.

Requirements informed by the work domain dimensions include: support for common agile web-application development activities; and integration with common tools used to design and develop software. Agile has gained popularity as a software development method in large organizations (Barlow et al., 2011). All three organizations were adopting the agile process in their software development process. A project wiki site should, therefore, support the agile development process. Additionally, software design teams use various tools to support their work. The common tools used by the five project teams included email, a bug tracking system, project management tools, subversion control software, a file-sharing server, and graphic design applications. A project wiki site was not used to replace any of these commonly used tools, but rather to link artifacts created and/or stored in these various tools into a single location. Thus, the wiki technology should easily be integrated with common tools used in software design projects.

Another requirement informed by the work domain dimension is that the wiki technology should allow project teams to have multiple levels of access control on their project wiki sites. Like many web-application design projects in a large organization, the five project teams work with many stakeholders, internal and external. They needed to communicate and share some project information with different stakeholders. For example, Project Team D had several documents that were confidential to only the project team. Because the project wiki site is open to all Flynow employees, the project team shared those confidential documents via email rather than on the project wiki site. On the other hand, Project Team E had a wiki page that was created specifically for several senior executives who were the project sponsors. Therefore, the wiki technology should provide project teams the ability to control who can view or edit individual wiki pages.

Requirements informed by the social organization dimensions include: fostering cross-disciplinary collaboration among project members; and providing a platform for discussion and negotiation. All the five project teams were cross-functional teams representing multiple disciplines. The common disciplines represented in a web application design project are management, design, software engineering, and domain experts. A project wiki site could be structured to support frequent collaboration among different disciplines such as that between project managers and other disciplines and that between designers and software engineers, as presented in Figure 5-6. The project wiki site should be structured to also supportive of both types of collaboration found in this study, which are contributing and challenging. That is, the project wiki site should enable a project team to easily contribute their knowledge as well as to negotiate over their work.

Requirements informed by the activity dimension include: providing easy access to project information; the capability of finding and sharing various types of artifacts; and supporting all levels of sharing. All the five project teams used their project wiki sites as a central information space where they could easily find all project information. The findings reveal that the project teams shared many kinds of information artifacts, which differ in content, form, and format. Therefore, in order for a project wiki site to provide access to all project information, the wiki technology needs to enable project members to share and find various types of artifacts. For example, in addition to providing easy-editing capability, the wiki technology should provide a good search feature in order to support finding information stored on project wiki sites. Moreover, to meet information-sharing needs, the project teams engaged in all four levels of sharing, which are materialistic sharing, syntactic sharing, semantic sharing, and pragmatic sharing. The wiki technology should allow project members to transport artifacts, transfer artifacts, translate artifacts, and transform artifacts through a project wiki site. Specific functional requirements to support each level of sharing are discussed as part of implications for wiki design and development in Section 7.3.1.

Requirements informed by the actor dimension include: easy to learn; easy to use; flexible; and open. The members of the five project teams came in with different technical expertise and wiki skills. Twenty-four participants had little-to-no wiki experience before they started using their project wiki sites. Therefore, a project wiki site should not only be easy to use for non-technical users, but it should also be easy to learn for non-wiki users. Flexibility and openness are common attributes of most wiki platforms. The findings also show that these attributes contributed to user's positive attitude toward the use of wiki. Thus, a project wiki site for an agile software design project should be configured in a way that it remains flexible and open.

7.2 Discussion of Findings

The findings of this dissertation enhance our understanding of how software design teams use wikis to support information sharing across disciplines. By understanding the use of wikis in software design, we can advance our understanding for information sharing across disciplines, the effectiveness of wikis as an information sharing tool, and the enhancement of wikis to support interdisciplinary design. This section discusses these topics.

Information sharing across disciplines: Previous research in information sharing in design has focused on one particular level of sharing. For example, many studies on the creation and use of shared information spaces have focused on accessing shared artifacts (Bannon & Bødker, 1997; Carstensen, 2000; Lahti et al., 2004; Trigg et al., 1999) while other studies in design communication have focused on communicating the meaning of shared artifacts (Bucciarelli, 1994; Grudin & Poltrock, 1989; Sonnenwald, 1995; Yamaoka et al., 1998). In contrast, this dissertation investigates information sharing from a holistic perspective using the 4-T framework in order to account for multiple levels of sharing.

The empirical findings from all five software design teams show that design teams often engage in multiple levels of sharing when they share artifacts across disciplines through project wikis. For example, in the case of sharing a design specifications document, designers linked the document on a project wiki site, and provided a descriptive link label so that project members from other disciplines could easily find the document. The designers also made the design specifications document easily viewable by creating it as wiki content so that the document could be viewed on a project wiki site using any Web browser. To create a common understanding of the design specifications document, the designers posted clarifications and rationales throughout the document so that other project members would have similar interpretations. Finally, to hand off the design specifications to the developers, the designers added status information on a wiki page so that the developers knew which part of the document was ready for development. Thus, the findings show that successful sharing is not as simple as giving access to artifacts or communicating the meaning of shared artifacts, but rather successful sharing involves multiple sharing activities, which include transporting (sharing access), transferring (sharing content), translating (sharing meaning), and transforming (sharing use).

The effectiveness of wikis as an information-sharing tool: Wikis have been known to support software documentation (Aguiar & David, 2005; Correia, 2010; Xiao et al., 2007) and software requirements engineering (Decker et al., 2007; Ferreira & Silva, 2009; Wu et al., 2009; Yang, 2009). The findings of this dissertation show that software design teams used wikis to share all kinds of artifacts in order to support various design tasks. As a project information repository, participants reported that in their design settings wikis encouraged more information sharing. Because of the simplicity and ease of use of wikis, design team members felt that they were more willing to share and to consume shared information and that wikis made shared information more visible and easily accessible. This specific finding supports other studies on project wikis, and provides empirical evidence for many of the positive, but previously unsupported, claims about wikis (Bean & Hott, 2005; Danis & Singer, 2008; Munson, 2008).

Additionally, the findings show that wikis support materialistic sharing and syntactic sharing, with a potential to support semantic sharing and pragmatic sharing. That is, software design teams primarily used wikis to give access to and to deliver content of shared artifacts. While the findings reveal that the software design teams attempted to use wikis to discuss the meaning of shared artifacts and to enable the use of share artifacts, the teams appeared to have difficulty utilizing features that would enable these forms of information sharing. For example, while Wikipedia contributors tend to communicate via dedicated discussion spaces (Kittur et al., 2008; Viégas et al., 2004; Viégas et al., 2007), none of the software design teams in this study used the discussion space, one of wiki's key features. The software design teams felt that the feature was too separate from the content of shared artifacts the teams would like to discuss.

The enhancement of wikis to support information sharing: Previous research indicates that the use of corporate wikis is influenced by various factors such as user expectations, the nature of information, existing information technology and practices, and corporate culture (Danis & Singer, 2008; Arazy & Croitoru, 2010; Grudin & Poole, 2010; Holtzblatt et al., 2010; Stocker & Tochtermann, 2009; White & Lutters, 2009; Yates, 2010). Similarly, the findings of this dissertation show that the use of project wikis was influenced by organizations, design projects, project teams, information sharing practices, and individual project members' skills and attitude. Additionally, this dissertation not only identifies factors influencing the use of project wikis (discussed in Chapter 5, Section 5.3), but also illustrates how the findings could inform system requirements for enhancing wikis.

The software development community has been interested in developing wiki clones, augmenting the wiki technology, and incorporating wiki software with other tools, to accommodate specific needs in different contexts (e.g., Aguiar & David, 2005; Hanrahan et al., 2011; Rankin et al., 2009). However, it is interesting to note that many development efforts were not informed by research into the use of wikis, disconnecting empirical research from wiki design and development. Guided by the CWA framework, this dissertation attempts to bring the empirical findings of wiki use to system design for wikis, similar to the study Fidel et al. (1998). The system requirements for wikis to better support information sharing in software design teams (presented in Table 7-1) could be used to inform both wiki adoption and wiki design.

Credibility and validity of findings

As with all qualitative research, this dissertation faces the challenge of credibility and validity. The quality of qualitative research can be judged based on its credibility and validity (Patton, 2002). Credibility refers to the credibility of both the researcher and the study results; how credible or convincing they are. Validity refers to the validity of the data; how truthful the data are in answering the research questions.

To address credibility, I employed two strategies. First, throughout the data analysis phase, I reviewed and discussed my coding scheme and preliminary results with my peers (PhD candidates in the Information School) and my dissertation committee members. I also presented the preliminary results to various research venues such as the Wikis for Software Engineering Workshop, the International Symposium on Wikis and Open Collaboration, the doctoral colloquium at GROUP conference, and the Special Interest Group on Design of Communication conference. Second, to increase the credibility of researcher, I reported any personal and professional information that had affected data collection, analysis, and interpretation. For example, I reported that I was involved in Project A and Project B as a part-time designer while I was collecting data. I also reported the imbalanced in data collected at three research sites, and how my analysis accounted for the imbalance.

To address validity, I employed two strategies. First, I triangulated data collection methods by using multiple methods, including observations, interviews, and wiki content and log review. The use of multiple methods collecting different types of data from different sources has provided cross-data validity. Second, I asked key participants from each study site to review preliminary findings as a way to verify the accuracy and completeness of my interpretation.

7.3 Practical implications

7.3.1 Implications for wiki technology

Implications for wiki technology are discussed from two aspects: wiki adoption and wiki design. The implications for wiki adoption are aimed at organizations and project teams that would like to use wiki technology to support software design and development projects. The implications for wiki design aim at practitioners who design and/or develop wiki technology.

Implications for wiki adoption

A number of success factors for enterprise wiki adoption have been identified, and design recommendations for enterprise wikis have been proposed such as aligning user expectation, nature of information, information quality, existing information technology and practices, and corporate culture (e.g., Cole, 2009; Danis & Singer, 2008; Grudin & Poole, 2010; Holzblatt et al., 2010). Implications of this dissertation aim specifically at the adoption of wiki technology for software design and development projects.

Implications for organizations

The findings have four significant implications for organizations that plan to adopt wiki technology for their software design and development projects. First, many organizations might benefit from grassroots efforts to adopt wiki technology as similar to all three organizations in this study. The grassroots-adoption of wiki technology that spreads beyond an IT department will increase familiarity with the wiki technology in daily work, which could potentially reduce time and effort to learn how to use a project wiki site. Additionally, a bottom-up approach like the grassroots effort might generate a positive attitude among employees toward the wiki technology than if it were mandatory from management. Second, high-level management could provide support to encourage project teams to use a wiki site as a project information repository. Third, central IT support for configuration and maintenance of a wiki server could lead to successful wiki adoption because it releases project teams from technical work required to set up project wiki sites. Finally, a guideline for wiki site creation and maintenance could be developed to create a consistency across multiple wiki sites. The guideline should account for an entire life cycle of project wiki sites. This implication supports previous research that has recognized the importance of documenting clear guidelines and policies (Holzblatt et al., 2010).

Implications for projects

The findings also suggest practical implications for wiki adoption at a project level, including creating group buy-in, designing a project wiki site together, keeping a project wiki site organized, developing an attitude of "our wiki site," and recommendations for supporting each of four information sharing levels (materialistic sharing, syntactic sharing, semantic sharing, and pragmatic sharing).

Creating group buy-in. Group buy-in and positive attitudes were found to contribute greatly to the success of project wiki sites. Therefore, a project team should start by generating buy-in from the project members, in particular from a project manager because a project wiki site could significantly benefit project management and communication. A project manager could provide a positive influence on the wiki use for other project members. For example, the project managers in Project C and Project D not only used the project wiki sites themselves, but also encouraged other project members to use the project wiki sites on a regular basis. The project manager of Project D, in particular, provided one-on-one training to the project members who had little wiki experience.

Designing a project wiki site together. The team members are the primary users of a project wiki site. Therefore, project members should design a project wiki site together so that their expectations are aligned and their various needs are incorporated. It is important that project teams understand the needs of all project members, particularly those who come from different disciplines. While designing a project wiki site, the project team members can discuss what information is expected on the project wiki and how information should be organized. Through this process, the project members are learning what is important to each of them, which is important for the future success of the project wiki operation.

Project members should also take into account various influential factors such as cultural differences between disciplinary groups, existing workflow, existing tools, and the nature of shared artifacts. For example, if most team members already have multiple systems in place to document and share information like Project Team C and D, the project team could use a project wiki site as a portal to provide a central access to the shared artifacts from different places. If project members are willing to change their documentation workflow and most artifacts do not require advanced formatting, the project team could use the project wiki site as a place to create and share artifacts.

Keeping a project wiki site organized. One of the wiki design principles is *organic* (Leuf & Cunningham, 2001). That is, the content and the structure of a wiki site are expected to grow organically, which has worked well for Wikipedia and like sites, but it may not work best for enterprise wiki sites (Díaz & Puente, 2011). The findings of this dissertation pointed out that some pre-defined structure was preferred by software design project teams because it was useful for members in adding and finding information. However, there was a need for balance between too much structure and too much flexibility in a project wiki site. A project team could provide a structure to the project wiki site without losing flexibility by avoiding a highly structured template for the project wiki site that has pre-defined links for future content. Rather a project team could create a loosely structured project wiki page template that could be adapted to meet the needs of individual projects. The project team should be open to changes and reorganize the project wiki site as needed. It will also be helpful to have at least one wiki advocate in the project to occasionally tidy-up the project wiki site (similar to the gardener role on Wikipedia).

Developing an attitude of "our wiki…not mine, not yours." A project manager or wiki advocates should encourage project teams to take ownership of an entire wiki, not just the content they have created. A project team should take a full advantage of the wiki's open editing and roll-back features to make changes to any content on the project wiki site. It should be emphasized that all changes could be easily reverted. Therefore, project team members should not be afraid of revising or editing any content. In addition, project teams should feel comfortable overwriting existing content when revising rather than creating multiple versions of the same content. This form of team ownership could emerge as teams become more experienced with the wiki technology and agile development.

Recommendations for specific information-sharing levels. Software design and development project teams can carry out the following suggestions for information-sharing practices in order to maximize the wiki's capability to support each information-sharing level.

To support materialistic sharing, project teams can develop a naming scheme for wiki pages as part of designing their project wiki site to help project members easily find information they are looking for. With a good naming scheme, project members can quickly identify relevant pages from Recent Changes, Watch List, Search Results, and Index pages. Based on the findings, a recommendation is to be consistent and always include a project name in a wiki page title like "[Project name]_[document title]." In addition to a naming scheme, project teams can make use of the tagging mechanism provided by some wiki platforms, such as a category namespace in MediaWiki and a tag in ConfluenceWiki, in order to create an index page for project-related wiki pages. Project teams can also create descriptive link labels to make materialistic sharing more successful. With the hyperlink mechanism, the wiki technology provides an easy, flexible way to label links posted on a wiki page. A link can be labeled with as much description as a project teams need to communicate what it links to. It could also be helpful to also include a type of artifact, whether it is an internal wiki page, a document file, an image, or an external website, so that users know what to expect when clicking the link.

To support syntactic sharing, project teams can create a glossary of terms to allow project members to describe their discipline-specific terms or vocabulary used on a project wiki to help project team members develop common language across disciplines. Previous research has shown that a common language can be useful for interdisciplinary design teams (Hertzum, 2004). It is particularly useful when project team members come from different organizational units, and are not familiar with each other's work. Another suggestion is to utilize the simplicity of wiki editing by keeping document formatting as simple as possible so that shared artifacts can be created in the wiki format. With reported formatting limitations of the wiki technology, shared artifacts that require advanced formatting should not be created in a project wiki site. If a shared artifact is created by a discipline-specific tool (such as Adobe Photoshop or MS Project), project teams should convert them into wiki viewable formats so that it does not require specific software applications to view.

To support semantic sharing, project teams can provide annotation of wiki content and utilize a discussion space provided in the wiki site. Annotation or explanation could be displayed with special visual treatment so that it is easily distinguished from the real content. In the software design context, annotation could be used to describe design rationales, which are critical for the design process (MacLean et al., 1998). Project teams could utilize a discussion space provided in the wiki site for cross-discipline discussions. Although all software design teams in the study preferred a face-to-face discussion, a discussion space in the wiki site could be used to prepare for face-to-face discussions or to support distributed collaboration.

To support pragmatic sharing, project teams can add status information to wiki documents and structure wiki content to make them easy to use. Project teams can use the status information to communicate with other project members, especially those who are in different disciplines, whether the information is ready to be used or not. Adding status information to wiki content can also reduce a concern that unfinished work would be misunderstood or used immaturely. The status information could be added to a wiki page itself, a link label, or a summary field when editing a wiki page. Moreover, project teams should discuss early on what information is needed by different disciplines, and how such information could be presented to support its use. It is important for the project members who create artifacts have a discussion with those who will be using the shared artifacts regarding how to structure the artifacts in a project wiki site to best support their use.

Implications for wiki design and development

In general, the findings suggest that the wiki technology should be designed to support various phases of design, various types of shared artifacts, and all levels of sharing, Some specific design recommendations for wiki technology can be drawn from the findings. Wiki designers and developers can carry out these recommendations to build wiki engines or to develop wiki extensions that facilitate information sharing and collaboration in software design and development. Organizations and project teams can also use these recommendations as a guideline when choosing the right wiki platform for design collaboration. The design recommendations are to:

Easily integrate with other collaborative tools. Software design teams employ many collaborative tools to create and share artifacts during the design process. A project wiki site did not replace any information-sharing or collaborative tools used by software. Instead, project teams used a project wiki site to connect various tools together. The wiki technology should be easily integrated with common tools used in software design and development such as bug tracking systems, issue tracking systems, and task management tools.

Support creation and presentation of various artifacts. Common formats for information shared on a wiki site are text, images, and links. To account for heterogeneous artifacts shared by software design teams, the wiki technology should allow users to create different types of artifacts, which differ in format and content, and usage. It should also provide ability to easily present artifacts in multiple formats, such as text, images, executable code, and multimedia.

Support materialistic sharing. Although the wiki technology makes it easy for users to share access to shared artifacts, there is room for improvement. Materialistic sharing through wiki technology could be improved by providing effective search functionality, and enabling multiple access control levels.

Support syntactic sharing. There are several ways to improve syntactic sharing through wiki technology. First, the wiki technology should provide a mechanism to easily convert different content into a wiki-presentable format. Second, a wiki's text editor should be designed to support complex formatting, while still allowing a simple text format. Third, the wiki technology should allow users to easily view various types of content within an Internet browser. Fourth, the wiki technology should provide the ability to create and reuse wiki content templates.

Support semantic sharing. Software design teams underutilized the discussion feature in wiki technology because they felt that the discussion was too separate from shared content itself. The software design teams added comments or questions on the content page rather than the designated discussion page provided in the wiki platform. Therefore, wiki technology should integrate a discussion space into the content page in forms of, for example, in-line comment, annotation, and page comment. This improvement will allow project members to create a feedback loop around artifacts shared on a project wiki site.

Support Pragmatic sharing. Software design teams used a project wiki site to create a shared task list when multiple members collaborated on the same task. An option to make wiki edits update synchronously without users needing to refresh a browser can make the use of a shared task list more efficient. Another suggestion to improve pragmatic sharing relates to handing off shared artifacts. Because the wiki technology allows work-in-progress to be accessible by all users, the wiki technology should provide a way to easily communicate when the shared artifacts are ready for use. Possible options are a document status setting or a push notification.

7.3.2 Implications beyond the wiki technology

Although this dissertation research focused on wiki technology, its findings provide broader implications beyond the wiki technology. Implications can be drawn for interdisciplinary design collaboration, distributed design collaboration, and agile development.

Implications for interdisciplinary design collaboration

A number of implications regarding information sharing and collaboration in interdisciplinary design can be derived from the findings based on an investigation of wiki use by software design teams. First, information sharing across disciplines is crucial to interdisciplinary design collaboration, and occurs throughout the design process. Thus, design teams need to recognize its importance and plan to support interdisciplinary sharing early in the design process. For instance, design teams should discuss their information-sharing plan in a project kick-off meeting, during which they could discuss what information is needed by different disciplines, how they would share discipline-specific information, and what information-sharing tools are available. As the information needs may change overtime, design team members should continue to discuss their information-sharing plan throughout a project life cycle.

Second, all four levels of information sharing are equally important for successful interdisciplinary design collaboration, and should be supported by collaborative tools. Design team members may give more attention to materialistic sharing, syntactic sharing, and semantic sharing. Pragmatic sharing is crucial in design collaboration, because team members often require artifacts shared by other disciplines in order to complete their task (e.g., software developers need design specifications from interaction designers in order to code). However, pragmatic sharing may not get as much attention as other levels, especially when choosing information-sharing tools. It is important to understand not just what information is needed by other team members but also how they need the information to be presented so that it is easy to use. For example, interaction designers should discuss with software developers and QA engineers how design specifications should be created to meet their needs.

Third, interdisciplinary design teams should choose collaborative tools based on types of shared information, information-sharing needs, and information-sharing processes. Many collaborative tools focus on sharing specific types of information, or support specific levels of sharing such as an email application, a file-sharing system, and an instant chat application. Interdisciplinary design teams should take into considerations types of shared artifacts, tasks, and information-sharing levels when choosing collaborative tools to support information sharing for their projects. If design teams anticipated a lot of framing artifacts, the study findings suggest that they consider tools that can support semantic sharing because a shared understanding across disciplines is important when sharing framing artifacts. On the other hand, if design teams anticipate a lot of procedural artifacts such as a project schedule and a shared to-do list, they could consider using other tools that better support pragmatic sharing than a wiki site.

Finally, the findings of this dissertation provide empirical evidence that the wiki technology can be used to support information sharing in interdisciplinary design. It has long been recognized that design team members bring with them disciplinary differences such as different perspectives, different work processes, different communication, and different tools (Bucciarelli, 1994; Fischer, 2001; Sonnenwald, 1995). In a large organization, where many interdisciplinary design teams are situated, these differences can be even greater. Thus, information sharing across disciplines can be even more challenging. Current collaborative tools do not fully support interdisciplinary collaboration (Adamczyk & Twidale, 2007; Haesen et al., 2009). The wiki technology, when configured and used properly, can lead to effective information sharing and successful collaboration. The simplicity and flexibility will encourage design team members to share more information as well as access shared information more frequently. With wiki technology, project work becomes more visible across disciplines; design team members can view shared information easily.

Implications for distributed design collaboration

Large interdisciplinary design projects, particularly software development projects, often involve distributed teams who are from different organizations, different physical locations, or even different time zone. Many studies on distributed design teams have indicated that communication and collaboration among team members who are geographically dispersed are major challenges (Happel et al., 2008; Larsson, 2003; Walthall et al., 2011). Information sharing becomes very crucial, but challenging, to distributed teams, even more so than collocated teams.

The implications for interdisciplinary design collaboration described in the previous section not only apply to collocated teams, but can also apply to distributed teams as well. The types of shared artifacts, information-sharing needs, and multiple levels of information sharing identified in this dissertation will likely be applicable to distributed design teams. While the findings show that collocated teams prefer to share information in person, in particular semantic sharing (e.g., showing a wiki content in a meeting, having a face-to-face discussion), information-sharing processes of distributed teams must always be mediated by collaborative technologies due to their geographical distance. Therefore, it is important for distributed teams to choose tools that will support transporting, transferring, translating, and transforming processes, both synchronously and asynchronously.

Wiki technology has been known to benefit distributed design teams, particularly on project management and software documentation (Clerc et al., 2010; Wei et al., 2005; Xiao, 2007). With the wiki technology, team members can easily access shared information and collaboratively write design documentation without the need for special HTML knowledge or special tools. However, several lessons learned regarding the use of wiki technology by collocated teams should be considered when distributed teams plan to use the wiki technology. First, although the wiki technology quite effectively supports asynchronous interactions among team members such as co-editing, it has some limitations on supporting synchronous interactions. Changes made on a wiki site take effect as soon as wiki contributors save them. The changes, however, will not appear instantly on the other end until wiki consumers refresh a browser. A recommendation for distributed teams is to make each other aware of their current actions via other communication tools.

Second, the relationship among shared artifacts and their change history was found to be important to distributed software development (Vivian et al., 2011); distributed teams can increase awareness of the relationship artifacts through a project wiki by developing a naming scheme for wiki pages and artifacts and using a descriptive link labels. They should also provide a description or summary of changes for any revision they make on a wiki page or a file. Finally, because distributed teams have fewer opportunities for direct interaction among team members, they should make an effort to promote semantic sharing through a project wiki site by creating a glossary of important terms, annotating comments and feedback, posting questions on a wiki page, and utilizing a discussion space for team discussion. This will create a shared understanding of content and artifacts shared on a project wiki site.

Implications for agile development

Because all participating organizations recently adopted the agile development approach, implications for agile development could be derived from these dissertation findings in terms of information-sharing practices and wiki use in agile development. Agile development is based on iterative and incremental development, which involves adaptive planning, evolutionary development and delivery, and a time-limited iterative approach. It encourages open, frequent communication among team members, which results in an increase in information sharing across disciplines. Additionally, agile teams are encouraged to work on a smaller set of features from the beginning to the end rather than to work on each layer of a whole application (i.e. business requirements, user interface, database, code, and testing). This approach is known as vertical slicing. As a result, agile development projects present a dynamic environment for information sharing across disciplines, where information sharing occurs more frequently, in particularly pragmatic sharing.

Agile project members are normally collocated, and often have a daily team meeting, during which project teams give status updates. So, when compared to traditional development approaches, agile development approaches are likely to have more direct communication and less documentation. In addition, according to the agile manifesto[9] the agile approach favors working software more than documentation. That is, agile development teams should reduce their documentation efforts so that they can focus on coding software. Previous research, however, has shown that face-to-face communication is efficient and effective at conveying information among project members, and that agile practitioners need documentation (Stettina & Heijstek, 2011). Through my research, I also found that agile (and agile want-to-be) teams spent substantial effort on documentation. Four out of five project teams spent substantial time and efforts on documentation, particularly getting documents into the right format. Formatting seems important and inevitable because it not only makes shared documents readable, but also can make meaning clearer as well as can make them easily usable.

Considering wiki design principles (Leuf & Cunningham, 2001) and agile manifestos (http://www.agilealliance.org/the-alliance/the-agile-manifesto/) as shown in Figure 7-2, the wiki technology and agile approach share some goals. They both strive for simplicity, flexibility, and open collaboration.

[9] The manifesto for agile software development is developed by the Agile Alliance to guide software development, methodologies, and organizations in more agile ways.
[Source: http://agilemanifesto.org/]

Wiki design principles		Agile manifestos
• Simple • Open • Incremental • Organic • Mundane • Universal	• Overt • Unified • Precise • Tolerant • Observable • Convergent	• Individuals and interactions over processes and tools • Working software over comprehensive documentation • Customer collaboration over contract negotiation • Responding to change over following a plan

Figure 7-2: Wiki design principles vs. Agile manifestos

Some researchers have investigated the use of wiki technology to support agile development, and have suggested that the wiki can be utilized to support agile development teams (Aguiar & David, 2005; Ferrerira & Silva, 2009). In contrast, this dissertation provides insight into how wikis can help organizations transition from traditional development to agile development. First, the wiki technology encourages project teams to share more information with its simplicity and open access. Project members can easily add information to a wiki site from a Web browser. They can also easily access information shared on a wiki site from any Web browser without needing special tools. Thus, a project wiki has a low barrier to sharing information, and project work is more visible across disciplines. Second, it encourages project teams to keep documentation simple with its wiki syntax markup. Most wiki platforms offer few formatting options, which forces project teams to keep documentation to minimal formatting. Finally, it encourages project teams to respond to changes quickly by making it easy to change documentation. Project teams may feel more comfortable making changes at any time because most wiki platforms provide revision history and revert. If any mistake is made, it can be recovered quickly. Project teams may not fully adopt an agile development approach, but by adopting the wiki technology as their common information-sharing tool they are stepping closer to being agile.

7.4 Theoretical Contributions

7.4.1 Information behavior

The dissertation contributes to the study of information behavior by expanding our understanding of information-sharing behavior in highly collaborative contexts such as software design projects. Previous research on information sharing has examined information sharing as individual information sharing behavior such as information sharing as a consequent of information seeking or information giving (Erdelez & Rioux, 2000; Marshall & Bly, 2004; O'Day & Jeffries, 1993; Savolainen, 2007). Existing research often focused on particular levels of information sharing (i.e. sharing artifacts, sharing a message, sharing meaning, and sharing knowledge), which did not capture a holistic view of information-sharing behavior, where people may engage in all levels of sharing when they share information.

This dissertation is a step toward investigating information sharing as collaborative information behavior. It applies a multi-dimensional approach by combining two conceptual frameworks, 3-T framework and CWA, in order to gain a more holistic understanding of information sharing in software design teams. The analysis of empirical data provides a deeper understanding of the information sharing during interdisciplinary design processes by identifying (1) information types shared across disciplines; (2) information-sharing needs; and (3) common information-sharing processes.

There are a number of research studies that have investigated the information behavior of professionals, and identified information needs by professionals. However, those identified needs are largely related to information-seeking behavior (e.g., Bruce et al., 2003; Fidel & Green, 2004; Pinelli, 1991; Savolainen, 2007; Yitzhaki & Hammershlag, 2004). This dissertation, in contrast, reveals information needs that motivate information sharing across disciplines during a design process.

This dissertation provides a methodological framework for developing future empirical studies on information sharing in design and other collaborative settings. It demonstrates how these two frameworks are complimentary, and how together they can be used to investigate the use of collaborative technologies to support information sharing by interdisciplinary design teams. While the 4-T framework (extending the 3-T framework) allows researchers to capture the complexity of information-sharing activities, the CWA framework allows researchers to account for the context, in which the information sharing occurs. The 4-T framework can be used as a conceptual model that is useful for a designing research process and interpreting findings.

7.4.2 Knowledge management

The dissertation enhances the knowledge management theoretical framework (the 3-T framework), and expanded its application to a different field. The 3-T framework was originated and applied in the knowledge management field (Carlile, 2002). I assessed the applicability of Carlile's 3-T framework for explaining information sharing among disciplinary groups that used wikis during the design process. The dissertation also contributes to the work on cross-boundary coordination, where knowledge management has been noted as a major challenge (Bechky 2003; Carlile 2002; Dougherty 1992; Kellogg et al., 2006). Previous research on cross-boundary coordination had only touched on the intangible aspect of knowledge, including transferring knowledge with a common language, translating knowledge with shared understanding, and transforming knowledge with integrative knowledge. This dissertation has shown that cross-boundary coordination may involve a tangible aspect of knowledge, particularly if we consider information items as a representation of codified knowledge or as explicit knowledge (Polanyi, 1983). Thus, the 4-T framework has one more layer, transport, which was not included in the 3-T framework to account for a tangible aspect of knowledge sharing. Additionally, throughout this dissertation, I operationalized information sharing processes in each layer and identified common sharing processes for information sharing supported by the wiki technology as presented in Chapter 6. This dissertation has shown that the 4-T framework can provide a more integrative perspective that expands our models for information-sharing processes.

7.4.3 Wiki research

This dissertation also contributes to the wiki research community. Although wiki technology has caught a lot of attention from both academia and industry since the year 2000, the scientific literature on wiki technology has focused heavily either on technological development of wikis and around Wikipedia. The number of studies investigating the use of wikis in a corporate context is still relatively low comparing to the studies on Wikipedia. Existing research has already found that corporate wikis are different from public wikis in their nature and needs (Danis & Singer, 2008, Grudin & Poole, 2010; Majchrzak et al., 2006). The findings of this dissertation, which investigated software design project wiki sites, a subset of enterprise wiki sites, echoed the differences identified in previous research, including a need for complex access control, no vandalism issues, and a need for more structure.

This dissertation provides empirical evidence on the use of wikis in software design work, which has been noted as one of the most common work activities supported by the use of wikis (Majchrzak et al., 2000). Previous research on the use of wikis in software design and development are more system-development focused, and have focused on particular aspects of design collaboration such as software documentation, requirement engineering, and programming (e.g., Aguiar & David, 2005; Ferrerira & Silva, 2009; Xiao et al., 2007). This dissertation, in contrast, investigates the use and user experience, focusing on the information sharing, an integral part of design collaboration that spans the project lifecycle. The findings provide a better understanding on how software design teams use wikis to support information sharing across disciplines throughout the design project. The findings also provide empirical evidence for many of positive, but previously unsupported, claims about wiki technology such as its simplicity, ease of use, and support for information sharing. These claims hold true in the context of software design collaboration. In addition, the dissertation offers wiki researchers a methodology to evaluate the use of wikis in various contexts by using the 4-T framework along with the CWA framework to guide data collection and data analysis.

7.4.4 Design research

This dissertation examines design collaboration from information sharing perspectives. Its findings have broadened our understanding of information sharing needs during the design process and how collaborative technologies can be developed particularly to support information sharing across disciplines, which is crucial to interdisciplinary collaboration. Based on my review of literature in Chapter 2, the literature in information systems in design has primarily focused on supporting the sharing of information artifacts, which is only one aspect of sharing. This dissertation has offered an alternative conceptual foundation to account for information sharing when investigating collaborative design work; it emphasizes that multiple levels of sharing should be considered. This dissertation also provides a methodology to evaluate collaborative technologies with the aim to generate recommendations for technology adoption and improvement. The 4-T framework can be used to evaluate how collaborative technologies support various levels of information sharing while the CWA can be used to analyze the context of technology use to understand its impact.

7.5 Conclusions

7.5.1 Limitations

It is important to note the limitations of this dissertation. First, the sample size was small, and the study sample may not be a representative of all interdisciplinary design teams. Thus, the findings may not be generalized beyond the sample. However, the small sample size was appropriate for an exploratory study because studying five software design teams in three organizations was sufficient to provide an insight on both information sharing practices and the use of wikis to support information sharing practices during interdisciplinary design processes.

Second, studying design teams in a real-world setting presents a limitation on data collection. Wiki log files were not available at two research sites. Access to the participants was limited during the research timeframe, which prevented me from conducting interviews with additional project members. Because this was an exploratory study, I collected as much data as I could. (i.e., in one project, I interviewed three project members from the same discipline). To overcome the unbalanced data collected from different research sites, I revised the data analysis to account for these differences.

Third, the 4-T framework presents a limitation as a conceptual framework used to investigate information sharing because, for the scope of this dissertation, I only investigated the sharing of tangible information on project wikis. Therefore, the 4-T framework may not explain the sharing of intangible artifacts such as ideas and thoughts or the sharing processes in other information-sharing tools. Additionally, with my efforts to operationalize the information sharing processes, the distinction between information sharing levels can still be fuzzy. For instance, having discussion can be defined as semantic sharing and/or pragmatic sharing depending on the goal of the discussion. If the discussion aims at clarifying the meaning of artifacts, it can be considered semantic sharing. On the other hand, if the discussion aims at negotiating the use of artifacts, it can be considered pragmatic sharing.

Finally, one common problem of qualitative field study is that observer effect could affect the data collected through participant observation at the EdApp group, the State University. Because I had worked with the EdApp group for 2 years prior to conducting the data and the period of study was over a year, the effect of my presence as an observer was very minimal. In fact, all project team members thought of me as a designer rather than a researcher.

7.5.2 Recommendations for future research

This dissertation unfolds many aspects of information-sharing behavior in interdisciplinary design teams, yet it brings opportunities for future research in this area. It would be useful to study different types of interdisciplinary design teams to find out if their information-sharing needs and processes are similar across various design settings, or if identified behavior is very specific to the software design teams that were studied. Because this dissertation focused on information sharing through project wiki sites, it would be interesting to apply a similar method to examine information sharing through other collaborative technologies. The results of such studies can confirm the utility of methodological frameworks used in this dissertation.

The findings of this dissertation shed some light on the use of software project wiki sites. Common wiki activities, contributions, and issues were identified. Future research can investigate the use of project wiki sites by different types of projects, which will allow researchers to generalize research results and implications. One of the key findings suggests that project teams need to create a balance between structure and flexibility of their project wiki sites. Future research can further investigate this issue to better understand where the balance is. How much structure do project teams need for a project wiki site? What makes project wiki sites flexible while maintaining necessary structure? My last recommendation for future research concerns with the use of project wiki sites after projects have been completed. Most project wiki sites contain useful information about developed products or services, which could be used during maintenance phase. The issue of making a project wiki site useful after the project was completed came up in the data, but was out of scope for this dissertation. Thus, future research can investigate a transition from a design-and-development phase wiki site to a post-development phase wiki site. What needs to be done to successfully transform a project wiki site to a product/service wiki site?

7.5.3 Closing Remarks

My dissertation has focused on the use of wiki technology for information sharing by interdisciplinary design teams. It was motivated by the challenges of information sharing in interdisciplinary collaboration and the popularity of wiki technology as an attractive collaboration tool. The dissertation aimed to answer the following three research questions: (1) How did the design projects share design-relevant information across disciplines?; (2) To what extent do wikis support information sharing and interdisciplinary design activities?; and (3) How could wikis be enhanced or utilized to better support information sharing in interdisciplinary design? To answer these research questions, I collected qualitative and quantitative data from five software design projects in three organizations.

The dissertation findings enhance our understanding of how the design projects share information across disciplines and how they use the wiki technology to support such sharing. I hope that my dissertation will help practitioners in a number of ways. First, the dissertation provides the wiki adoption recommendations for organizations and project teams. Second, it provides design recommendations for wiki designers and developers. Third, the dissertation also provides a better understanding of information sharing and the use of wiki technology for interdisciplinary design, distributed design collaboration, and agile development.

The dissertation findings contribute to the body of knowledge in the fields of information behavior, knowledge management, wiki research, and design research. The dissertation enhances our understanding of how design teams share information across disciplines and how they use wiki technology to support such sharing. On the other hand, this dissertation raises many questions for future research as outlined in the previous section. I hope that this dissertation offers researchers a methodology to investigate information sharing behavior as well as a methodology able to assess the use of wiki technology in other collaborative contexts.

REFERENCES

Ackoff, R. (1989). From Data to Wisdom.*Journal of Applied Systems Analysis.* **16**: 3-9.

Adamczyk, P. D. and Twidale, M. B. (2007). Supporting multidisciplinary collaboration: requirements from novel HCI education. *Proceedings of the SIGCHI conference on Human factors in computing systems*, San Jose, California, USA, ACM. 1073-1076.

Aguiar, A. and David, G. (2005). WikiWiki weaving heterogeneous software artifacts. *2005 International Symposium on Wikis*, San Diego, CA, United States, Association for Computing Machinery, New York, NY 10036-5701, United States. 67-74.

Antin, J. (2011). My kind of people?: perceptions about wikipedia contributors and their motivations *Proceedings of the 2011 annual conference on Human factors in computing systems*, Vancouver, BC, Canada, ACM.

Antin, J., Yee, R., Cheshire, C. and Nov, O. (2011). Gender differences in Wikipedia editing *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, Mountain View, California, ACM.

Arazy, O. and Croitoru, A. (2010). The sustainability of corporate wikis: A time-series analysis of activity patterns. *ACM Trans. Manage. Inf. Syst.* **1**(1): 1-24.

Arazy, O., Gellatly, I., Soobaek, J. and Patterson, R. (2009). Wiki deployment in corporate settings. *Technology and Society Magazine, IEEE* **28**(2): 57-64.

Arias, E., Eden, H., Fischer, G., Gorman, A. and Scharff, E. (2000). Transcending the individual human mind—creating shared understanding through collaborative design. *ACM Transactions on Computer-Human Interaction* **7**(1): 84-113.

Avram, G. (2006) At the Crossroads of Knowledge Management and Social Software. *The Electronic Journal of Knowledge Management* **4**, 1-10 DOI:

Bannon, L. J. and Bødker, S. (1997). Constructing Common Information Spaces. *European Conference on Computer Supported Cooperative Work (ECSCW'97)*, Kluwer. 81-96.

Baraldi, S., Bimbo, A. D. and Valli, A. (2006). Bringing the Wiki Collaboration Model to the Tabletop World. *Multimedia and Expo, 2006 IEEE International Conference on*. 333-336.

Barlow, J. B., Giboney, J. S., Keith, M. J., Wilson, D. W., Schuetzler, R. M., Lowry, P. B. and Vance, A. (2011). Overview and Guidance on Agile Development in Large Organizations. *Communications of the Association for Information Systems* **29**(Article 2).

Bauer, K., Fokaefs, M., Tansey, B. and Stroulia, E. (2009). WikiDev 2.0: discovering clusters of related team artifacts *Proceedings of the 2009 Conference of the Center for Advanced Studies on Collaborative Research*, Ontario, Canada, ACM.

Bean, L. and Hott, D. D. (2005). Wiki: A speedy new tool to manage projects. *Journal of Corporate Accounting & Finance* **16**(5): 3-8.

Bechky, B. A. (2003). Sharing meaning across occupational communities: The transformation of understanding on a product floor. *Organization Science* **14**(3): 312–330.

Berlin, L. M., Jeffries, R., O'Day, V. L., Paepcke, A. and Wharton, C. (1993). Where did you put it? Issues in the design and use of a group memory. *SIGCHI conference on Human factors in computing systems*, Amsterdam, The Netherlands, ACM Press. 23-30.

Bhatti, Z. A., Baile, S. and Yasin, H. M. (2011). The success of corporate wiki systems: an end user perspective *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, Mountain View, California, ACM.

Biuk-Aghai, R. P. and Lei, K. H. (2010). Chatting in the Wiki: synchronous-asynchronous integration *Proceedings of the 6th International Symposium on Wikis and Open Collaboration*, Gdansk, Poland, ACM.

Boisot, M. and Canals, A. (2004). Data, Information, and Knowledge: Have we Got It Right? *Journal of Evolutionary Economics* **14**.

Boland, R. J. J. and Tenkasi, R. V. (1995). Perspective Making and Perspective Taking in Communities of Knowing. *Organization Science* **6**(4): 350-372.

Bossen, C. (2002). The parameters of common information spaces: the heterogeneity of cooperative work at a hospital ward. *2002 ACM conference on Computer supported cooperative work*, New Orleans, Louisiana, USA, ACM Press. 176-185.

Boujut, J. F. and Blanco, E. (2003). Intermediary Objects as a Means to Foster Co-operation in Engineering Design. *Computer Supported Cooperative Work* **12**(2): 205-219.

Bruce, H., Fidel, R., Pejtersen, A. M., Dumais, S. and Grudin, J. (2003). A comparison of the collaborative information retrieval of two design teams. *The New Review of Information Behavior Research* **4**: 139-153.

Bucciarelli, L. L. (1994). *Designing engineers*, MIT Press.

Buckland, M. K. (1991). Information as thing. *Journal of the American Society for Information Science*(42): 351-360.

Burrow, A. L. (2004). Negotiating access within Wiki: a system to construct and maintain a taxonomy of access rules. *Proceedings of the fifteenth ACM conference on Hypertext and hypermedia*, Santa Cruz, CA, USA, ACM. 77-86.

Cannon-Bowers, J. A., Salas, E. and Converse, S. (1993). Shared Mental Models in Expert Team Decision Making. *Individual and Group Decision Making*. J. J. Castellan. Hillsdale, NJ, Earlbaum**:** 221-246.

Carlile, P. R. (2002). A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development. *Organization Science* **13**(4): 442-455.

Carlile, P. R. (2004). Transferring, Translating, and Transforming: An Integrative Framework for Managing Knowledge Across Boundaries. *Organization Science* **15**(5): 555-568.

Carstensen, P. (2000). Common Information Spaces: A few lessons learned about real-life archives and the structuring of information. *Workshop on Cooperative Organization of Common Information Spaces*, Lyngby, Denmark.

Chau, T. and Maurer, F. (2004). *Knowledge Sharing in Agile Software Teams*.

Citera, M., McNeese, M. D., Brown, C. E., Selvaraj, J. A., Zaff, B. S. and Whitaker, R. D. (1995). Fitting information systems to collaborating design teams. *Journal of the American Society of Information Science* **46**(7): 551-559.

Clarkson, J. and Eckert, C. (2005). *Design process improvement: a review of current practice*. London [U.K.], Springer.

Clerc, V., Vries, E. d. and Lago, P. (2010). Using wikis to support architectural knowledge management in global software development *Proceedings of the 2010 ICSE Workshop on Sharing and Reusing Architectural Knowledge*, Cape Town, South Africa, ACM.

Connelly, C. E. and Kelloway, E. K. (2003). Predictors of employees' perceptions of knowledge sharing cultures. *Leadership and Organization Development Journal* **24**(5): 294-301.

Cook, S. D. N. and Brown, J. S. (1999). Bridging Epistemologies: The Generative Dance Between Organizational Knowledge and Organizational Knowing. *Organization Science* **10**(4): 381 - 400.

Correia, F. F. (2010). Supporting the evolution of software knowledge with adaptive software artifacts *Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion*, Reno/Tahoe, Nevada, USA, ACM.

Correia, F. F., Ferreira, H. S., Flores, N. and Aguiar, A. (2009). Incremental knowledge acquisition in software development using a weakly-typed Wiki *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, Orlando, Florida, ACM.

Cross, N. (1984). *Developments in design methodology*. Chichester, New York, Wiley.

Curtis, B., Krasner, H. and Iscoe, N. (1988). A field study of the software design process for large systems. *Communications of the ACM* **31**(11): 1268-1287.

Danis, C. and Singer, D. (2008). A wiki instance in the enterprise: opportunities, concerns and reality *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, San Diego, CA, USA, ACM.

Davenport, T. H. and Prusak, L. (2000). *Working knowledge: How organizations manage what they know*. Boston, Harvard Business School Press.

Davis, J. G., Subrahmanian, E., Konda, S., Granger, H., Collins, M. and Westerberg, A. W. (2001). Creating Shared Information Spaces to Support Collaborative Design Work. *Information Systems Frontiers* **3**(3): 377-392.

Davis, R. C., Landay, J. A., Chen, V., Huang, J., Lee, R. B., Li, F. C., Lin, J., Morrey III, C. B., Schleimer, B., Price, M. N. and Schilit, B. N. (1999). NotePals: lightweight note sharing by the group, for the group. *SIGCHI conference on Human factors in computing systems: the CHI is the limit*, Pittsburgh, Pennsylvania, United States, ACM Press. 338-345.

Decker, B., Ras, E., Rech, J., Jaubert, P. and Rieth, M. (2007). Wiki-Based Stakeholder Participation in Requirements Engineering. *Software, IEEE* **24**(2): 28-35.

Désilets, A., Paquet, S. and Vinson, N. G. (2005). Are wikis usable? *Proceedings of the 2005 international symposium on Wikis*, San Diego, California, ACM. 3-15.

deSouza, C. R. B., Redmiles, D. and Dourish, P. (2003). "Breaking the code", moving between private and public work in collaborative software development. *2003 international ACM SIGGROUP conference on Supporting group work*, Sanibel Island, Florida, USA, ACM Press. 105-114.

Détienne, F. (2006). Collaborative design : managing task interdependencies and multiple perspectives. *Interacting with Computers* **18**(1): 1-20.

Di Iorio, A. and Zacchiroli, S. (2006). Constrained Wiki: an Oxymoron? *Proceedings of the 2006 international symposium on Wikis*, Odense, Denmark, ACM.

Díaz, O. and Puente, G. (2011). Wiki scaffolding: helping organizations to set up wikis *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, Mountain View, California, ACM.

Dougherty, D. (1992). Interpretive barriers to successful product innovation in large firms. *Organization Science* **3**: 179-202.

Ducasse, S., Renggli, L. and Wuyts, R. (2005). SmallWiki: a meta-described collaborative content management system. *Proceedings of the 2005 international symposium on Wikis*, San Diego, California, ACM. 75-82.

Eckert, C. and Clarkson, J. (2005). Introduction: The reality of design. *Design Process Improvement: A review of current practice*. J. Clarkson and C. Eckert. London [U.K.], Springer**:** 1-28.

Eckert, C., Maier, A. and McMahon, C. (2005). Communication in design. *Design Process Improvement: A review of current practice*. J. Clarkson and C. Eckert. London [U.K.], Springer**:** 232-261.

Egli, U. and Sommerlad, P. (2009). Experience report - Wiki for law firms *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, Orlando, Florida, ACM.

Erdelez, S. and Rioux, K. (2000). Sharing information encountered for others on the Web. *New Review of Information Behaviour Research* **1**(January): 219-233.

Faraj, S. and Sproull, L. (2000). Coordinating Expertise in Software Development Teams. *Management Science* **46**(12): 1554-1568.

Ferreira, D. and Silva, A. (2009). An Enhanced Wiki for Requirements Engineering. *Software Engineering and Advanced Applications, 2009. SEAA '09. 35th Euromicro Conference on*. 87-94.

Fichter, D. (2005). Intranets, Wikis, Blickis, and Collaborative Working. *Online*(September/October): 47-50.

Fidel, R., Davies, R., Douglass, M., Holder, J., and Hopkins, C. (1999). A visit to the information mall: Web searching behavior of high school students. *Journal of American Society of Information Science*, **50**(1), 24-37

Fidel, R. and Green, M. (2004). The many faces of accessibility: engineers' perception of information sources. *Information Processing & Management* **40**(3): 563-581.

Fidel, R. and Pejtersen, A. M. (2004). From Information Behavior Research to the Design of Information Systems: the Cognitive Work Analysis Framework. *Information Research* **10**(1): Paper 210.

Fischer, G., Grudin, J., McCall, R., Ostwald, J., Redmiles, D., Reeves, B. and Shipman, F. (2001). Seeding, Evolutionary Growth and Reseeding: The Incremental Development of Collaborative Design Environments. *Coordination Theory and Collaboration Technology*. G. M. Olson, T. W. Malone and J. B. Smith. Mahwah, NJ, Lawrence Erlbaum Associates**:** 447-472.

Fokaefs, M., Tansey, B., Ganev, V., Bauer, K. and Stroulia, E. (2010). WikiDev 2.0: Facilitating Software Development Teams. *Software Maintenance and Reengineering (CSMR), 2010 14th European Conference on*. 276-277.

Fong, P. K.-F. and Biuk-Aghai, R. P. (2010). What did they do? Deriving high-level edit histories in Wikis *Proceedings of the 6th International Symposium on Wikis and Open Collaboration*, Gdansk, Poland, ACM.

Gasson, S. (2005). Boundary-Spanning Knowledge-Sharing In E-Collaboration. *38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 8*, IEEE Computer Society. 245b.

Geiger, R. S. and Ribes, D. (2010). The work of sustaining order in wikipedia: the banning of a vandal *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, Savannah, Georgia, USA, ACM.

Geisler, C. and Rogers, E. H. (2000). Technological mediation for design collaboration. *IEEE professional communication society international professional communication conference*, Cambridge, Massachusetts, IEEE Educational Activities Department. 395-405.

Goecks, J. and Cosley, D. (2002). NuggetMine: intelligent groupware for opportunistically sharing information nuggets. *7th international conference on Intelligent user interfaces*, San Francisco, California, USA, ACM Press. 87-94.

Goel, V. and Pirolli, P. (1992). The structure of design problem spaces. *Cognitive Science: A Multidisciplinary Journal* **16**(3): 395-429.

Goodnoe, E. (2005). Wikis make collaboration easier [content-management software]. *InformationWEEK*(1053): 54-8.

Grace, T. P. L. (2009). Wikis as a knowledge management tool. *Journal of Knowledge Management* **13**(4): 64-74.

Grudin, J. and Poltrock, S. E. (1989). User interface design in large corporations: coordination and communication across disciplines. *SIGCHI conference on Human factors in computing systems: Wings for the mind*, ACM Press. 197-203.

Grudin, J. and Poole, E. S. (2010). Wikis at work: success factors and challenges for sustainability of enterprise Wikis *Proceedings of the 6th International Symposium on Wikis and Open Collaboration*, Gdansk, Poland, ACM.

Guzdial, M., Rick, J. and Kehoe, C. (2001). Beyond adoption to invention: Teacher-created collaborative activities in higher education. *Journal of the Learning Sciences* **10**(3): 265-279.

Haake, A., Lukosch, S. and Schümmer, T. (2005). Wiki-templates: adding structure support to wikis on demand. *Proceedings of the 2005 international symposium on Wikis*, San Diego, California, ACM. 41-51.

Haesen, M., Meskens, J., Luyten, K. and Coninx, K. (2009). Supporting Multidisciplinary Teams and Early Design Stages Using Storyboards *Proceedings of the 13th International Conference on Human-Computer Interaction. Part I: New Trends*, San Diego, CA, Springer-Verlag.

Halfaker, A., Kittur, A., Kraut, R. and Riedl, J. (2009). A jury of your peers: quality, experience and ownership in Wikipedia *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, Orlando, Florida, ACM.

Hanrahan, B. V., Weksteen, T., Kong, N., Convertino, G., Bouchard, G., Archambeau, C. and Chi, E. H. (2011). Mail2Wiki: posting and curating Wiki content from email *Proceedings of the 16th international conference on Intelligent user interfaces*, Palo Alto, CA, USA, ACM.

Hansen, S. and Rennecker, J. (2010). Getting on the same page: Collective hermeneutics in a systems development team. *Inf. Organ.* **20**(1): 44-63.

Happel, H.-J., Maalej, W. and Stojanovi, L. (2008). Team: towards a software engineering semantic web *Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering*, Leipzig, Germany, ACM.

Hattori, T. (2011). Wikigramming: a wiki-based training environment for programming *Proceeding of the 2nd international workshop on Web 2.0 for software engineering*, Waikiki, Honolulu, HI, USA, ACM.

Henderson, K. (1991). Flexible Sketches and Inflexible Data Bases: Visual Communication, Conscription Devices, and Boundary Objects in Design Engineering. *Science, Technology, and Human Values* **16**(4): 448-473.

Hendry, D. G. (2004). Communication functions and the adaptation of design representations in interdisciplinary teams. *2004 conference on Designing interactive systems: processes, practices, methods, and techniques*, Cambridge, MA, USA, ACM Press. 123-132.

Hertzum, Morten (2004). Small-Scale Classification Schemes: A Field Study of Requirements Engineering. In *Computer Supported Cooperative Work*, **13** (1) pp. 35-61.

Hester, A. J. (2010). Increasing collaborative knowledge management in your organization: characteristics of wiki technology and wiki users *Proceedings of the 2010 Special Interest Group on Management Information System's 48th annual conference on Computer personnel research on Computer personnel research*, Vancouver, BC, Canada, ACM.

Hirsch, C., Hosking, J., Grundy, J. and Chaffe, T. (2010). ThinkFree: using a visual Wiki for IT knowledge management in a tertiary institution *Proceedings of the 6th International Symposium on Wikis and Open Collaboration*, Gdansk, Poland, ACM.

Holtzblatt, L. J., Damianos, L. E. and Weiss, D. (2010). Factors impeding Wiki use in the enterprise: a case study *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, Atlanta, Georgia, USA, ACM.

Horner, J. and Atwood, M. E. (2006). Design rationale: the rationale and the barriers. *the 4th Nordic conference on Human-computer interaction: changing roles (NordiCHI '06)*, ACM. 341-350.

Jaksch, B., Kepp, S.-J. and Womser-Hacker, C. (2008). Integration of a Wiki for Collaborative Knowledge Development in an E-Learning Context for University Teaching *Proceedings of the 4th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society on HCI and Usability for Education and Work*, Graz, Austria, Springer-Verlag.

Javanmardi, S., McDonald, D. W. and Lopes, C. V. (2011). Vandalism detection in Wikipedia: a high-performing, feature-rich model and its reduction through Lasso *Proceedings of the*

7th International Symposium on Wikis and Open Collaboration, Mountain View, California, ACM.

Jones, J. C. (1992). *Design Methods*. New York, Van Nostrand Reinhold.

Kane, G. C. and Fichman, R. G. (2009). The shoemaker's children: using wikis for information systems teaching, research, and publication. *MIS Q.* **33**(1): 1-17.

Kellogg, K. C., Orlikowski, W. J. and Yates, J. (2006). Life in the Trading Zone: Structuring Coordination Across Boundaries in Postbureaucratic Organizations. *ORGANIZATION SCIENCE* **17**(1): 22-44.

Kepp, S. J. and Schorr, H. (2009). Analyzing collaborative learning activities in wikis using social network analysis *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, Boston, MA, USA, ACM.

Kittur, A. and Kraut, R. E. (2010). Beyond Wikipedia: coordination and conflict in online production groups *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, Savannah, Georgia, USA, ACM.

Kittur, A., Suh, B. and Chi, E. H. (2008). Can you ever trust a wiki?: impacting perceived trustworthiness in wikipedia *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, San Diego, CA, USA, ACM.

Klobas, J. (2006). *Wikis: Tools for Information Work and Collaboration*, Chandos Publishing.

Kong, N., Convertino, G., Hanrahan, B. and Chi, E. (2011). VisualWikiCurator: a corporate Wiki plugin *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*, Vancouver, BC, Canada, ACM.

Kramer, M., Gregorowicz, A. and Iyer, B. (2008). Wiki trust metrics based on phrasal analysis *Proceedings of the 4th International Symposium on Wikis*, Porto, Portugal, ACM.

Kumar, V., Glicksman, J. and Kramer, G. A. (1994). A SHAREd Web to support design teams. 178-182.

Kussmaul, C. and Albert, S. (2007). Reading and writing with Wikis: progress and plans. *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*, Washington, DC, USA, ACM. 261-262.

Lahti, H., Seitamaa-Hakkarainen, P. and Hakkarainen, K. (2004). Collaboration patterns in computer supported collaborative designing. *Design Studies* **25**(4): 351-371.

Lam, S. K., Uduwage, A., Dong, Z., Sen, S., Musicant, D. R., Terveen, L. and Riedl, J. (2011). WP:clubhouse?: an exploration of Wikipedia's gender imbalance *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, Mountain View, California, ACM.

Langan-Fox, J., Anglim, J. and Wilson, J. R. (2004). Mental models, team mental models, and performance: Process, development, and future directions. *Human Factors and Ergonomics in Manufacturing* **14**(4): 331-352.

Larsson, A. T. P. o. t. i. A. (2003). Making sense of collaboration: the challenge of thinking together in global design teams. *Proceedings of the 2003 international ACM SIGGROUP conference on Supporting group work (GROUP '03)*, ACM. 153-160.

Larusson, J. A. and Alterman, R. (2009). Visualizing student activity in a wiki-mediated co-blogging exercise *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, Boston, MA, USA, ACM.

Leuf, B. and Cunningham, W. (2001). *The Wiki way : quick collaboration on the Web*. Boston, Addison-Wesley.

Levina, N. (2001). Multi-party information systems development: The challenge of cross-boundary collaboration. Cambridge, MA, Massachusetts Institute of Technology. **PhD**.

Liu, J. and Ram, S. (2011). Who does what: Collaboration patterns in the wikipedia and their impact on article quality. *ACM Trans. Manage. Inf. Syst.* **2**(2): 1-23.

Louridas, P. (2006). Using wikis in software development. *IEEE Software* **23**(2): 88-91.

Love, P. E. D. and Gunasekaran, A. (1997). Concurrent Engineering in the Construction Industry. *Concurrent Engineering* **5**(2): 155-162.

Löwgren, J. and Stolterman, E. (2004). *Thoughtful interaction design: a design perspective on information technology*. Cambridge, MA, MIT Press.

MacLean, A., Young, R. M. and Moran, T. P. (1989). Design rationale: the argument behind the artifact. *SIGCHI conference on Human factors in computing systems: Wings for the mind*, ACM Press. 247-252.

Mader, S. (2008). *Wikipatterns: A pratical guide to improving productivity and collaboration in your organization*. Indianapolis, Indiana, Wiley Publishing, Inc.

Majchrzak, A., Rice, R. E., King, N., Malhotra, A. and Ba, S. (2000). Computer-Mediated Inter-Organizational Knowledge-Sharing: Insights from a Virtual Team Innovating Using a Collaborative Tool. *Information Resources Management Journal* **13**(1): 44-53.

Majchrzak, A., Wagner, C. and Yates, D. (2006). Corporate wiki users: results of a survey. *2006 international symposium on Wikis*, Odense, Denmark, ACM Press. 99-104.

Malani, P. and Dwyer, C. (2005). Low Cost Collaborative Tools to Support Student Teams. *Students/Faculty Research Day, CSIS*. A1.1-A1.9.

Marshall, C. C. and Bly, S. (2004). Sharing encountered information: digital libraries get a social life. *4th ACM/IEEE-CS joint conference on Digital libraries*, Tuscon, AZ, USA, ACM

Press. 218-227.

McDonough, E. F. (2000). Investigation of factors contributing to the success of cross-functional teams. *Journal of Product Innovation Management* **17**(3): 221-235.

Meadow, C. T. and Yuan, W. (1997). Measuring the impact of information: defining the concepts. *Information Processing & Management* **33**(6): 697-714.

Mesmer-Magnus, J. R. and DeChurch, L. A. (2009). Information Sharing and Team Performance: A Meta-Analysis. *Journal of Applied Psychology* **94**(2): 535-546.

Mohammed, S., Klimoski, R. and Rentsch, J. R. (2000). The Measurement of Team Mental Models: We Have No Shared Schema. *Organizational Research Methods* **3**(2): 123-165.

Mola-Velasco, S. M. (2011). Wikipedia vandalism detection *Proceedings of the 20th international conference companion on World wide web*, Hyderabad, India, ACM.

Moreland, R. L. (1999). Transactive Memory: Learning Who Knows What in Work Groups and Organizations. *Shared Cognition in Organizations: The Management of Knowledge*. L. L. Thompson, J. M. Levine and D. M. Messick. Mahwah, N.J., L. Erlbaum**:** 3-31.

Munson, S. A. (2008). Motivating and enabling organizational memory with a workgroup wiki *Proceedings of the 4th International Symposium on Wikis*, Porto, Portugal, ACM.

Nelson, H. G. and Stolterman, E. (2003). *The Design Way*. Englewood Cliffs, New Jersey, Educational Technology Publications.

O'Day, V. L. and Jeffries, R. (1993). Information artisans: patterns of result sharing by information searchers. *conference on Organizational computing systems*, Milpitas, California, United States, ACM Press. 98-107.

Oxley, M., Morgan, J. T., Zachry, M. and Hutchinson, B. (2010). "What i know is...": establishing credibility on Wikipedia talk pages *Proceedings of the 6th International Symposium on Wikis and Open Collaboration*, Gdansk, Poland, ACM.

Palmer, C. L. (2001). *Work at the Boundaries of Science: Information and the Interdisciplinary Research Process*. Boston, MA, Kluwer Academic Publishers.

Panciera, K., Halfaker, A. and Terveen, L. (2009). Wikipedians are born, not made: a study of power editors on Wikipedia *Proceedings of the ACM 2009 international conference on Supporting group work*, Sanibel Island, Florida, USA, ACM.

Paquet, S. (2006). Wikis in business. *Wikis: Tools for Information Work and Collaboration*. J. Klobas, Chandos Publishing**:** 99-117.

Parker, G. M. (2003). *Cross-functional teams : working with allies, enemies, and other strangers* San Francisco, Jossey-Bass.

Parker, K. R. and Chao, J. T. (2007). Wiki as a Teaching Tool. *Interdisciplinary Journal of Knowledge & Learning Objects* **3**: 57-72.

Patton, M. Q. (2002). *Qualitative Research and Evaluation Methods*. Newbury Park, CA, Sage Publications.

Pejtersen, A. M., Sonnenwald, D. H., Buur, J., Govindaraj, T. and Vicente, K. (1997). The Design Explorer project: Using a cognitive framework to support knowledge exploration. *Journal of Engineering Design* **8**(3): 289-301.

Perry, M. and Sanderson, D. (1998). Coordinating joint design work: the role of communication and artefacts. *Design Studies* **19**(3): 273-288.

Pettigrew, K. E., Fidel, R. and Bruce, H. (2001). Conceptual frameworks in information behavior. *Annual Review of Information Science and Technology (ARIST)* **35**: 43-78.

Pinelli, T. E. (1991). The information-seeking habits and practices of engineers. *Science and Technology Libraries* **11**(3): 5-25.

Pinto, M. B., Pinto, J. K. and Prescott, J. E. (1993). Antecedents and Consequences of Project Team Cross-Functional Cooperation. *Management Science* **39**(10): 1281-1297.

Pirolli, P., Wollny, E. and Suh, B. (2009). So you know you're getting the best possible information: a tool that increases Wikipedia credibility *Proceedings of the 27th international conference on Human factors in computing systems*, Boston, MA, USA, ACM.

Polanyi, M. (1983). *The Tacit Dimension*. Magnolia, MA, Peter Smith.

Poltrock, S., Grudin, J., Dumais, S., Fidel, R., Bruce, H. and Pejtersen, A. M. (2003). Information seeking and sharing in design teams. *2003 international ACM SIGGROUP conference on Supporting group work*, Sanibel Island, Florida, USA, ACM Press. 239-247.

Poole, E. S. and Grudin, J. (2010). A taxonomy of Wiki genres in enterprise settings *Proceedings of the 6th International Symposium on Wikis and Open Collaboration*, Gdansk, Poland, ACM.

Potthast, M., Stein, B. and Gerling, R. (2008). Automatic Vandalism Detection in Wikipedia

Advances in Information Retrieval. C. Macdonald, I. Ounis, V. Plachouras, I. Ruthven and R. White, Springer Berlin / Heidelberg. **4956:** 663-668.

Prasad, S. and Tata, J. (2005). Publication patterns concerning the role of teams/groups in the information systems literature from 1990 to 1999. *Information & Management* **42**(8): 1137-1148.

Priedhorsky, R., Chen, J., Lam, S. K., Panciera, K., Terveen, L. and Riedl, J. (2007). Creating, destroying, and restoring value in wikipedia *Proceedings of the 2007 international ACM conference on Supporting group work*, Sanibel Island, Florida, USA, ACM.

Rader, E. (2009). Yours, mine and (not) ours: social influences on group information repositories *Proceedings of the 27th international conference on Human factors in computing systems*, Boston, MA, USA, ACM.

Radziwill, N. M. and Shelton, A. L. (2004). TWiki as a platform for collaborative software development management, Glasgow, UK, SPIE-Int. Soc. Opt. Eng. 609-17.

Raitman, R., Augar, N. and Zhou, W. (2005). Employing Wikis for Online Collaboration in the E-Learning Environment: Case Study. *Information Technology and Applications, 2005. ICITA 2005. Third International Conference on*. 142-146.

Rankin, J., Anslow, C., Noble, J., Chawner, B. and Gordon, D. (2009). Wikipublisher: a print-on-demand Wiki *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, Orlando, Florida, ACM.

Rasmussen, J., Pejtersen, A. M. and Goodstein, L. P. (1994). *Cognitive systems engineering*. New York, Wiley.

Reeves, B. and Shipman, F. (1992). Supporting communication between designers with artifact-centered evolving information spaces. *1992 ACM conference on Computer-supported cooperative work*, Toronto, Ontario, Canada, ACM Press. 394-401.

Restrepo, J. and Christiaans, H. (2004). Problem Structuring and Information Access in Design. *Journal of Design Research* **4**(2).

Reynolds, R. and Caperton, I. H. (2009). Comparison of middle school, high school and community college students' Wiki activity in Globaloria-West Virginia: (pilot year-two) *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, Orlando, Florida, ACM.

Rittel, H. W. J. and Webber, M. M. (1973). Dilemmas in General Theory of Planning. *Policy Sciences* **4**: 155-169.

Römer, A., Pache, M., Weihahn, G., Lindemann, U. and Hacker, W. (2001). Effort-saving product representations in design--results of a questionnaire survey. *Design Studies* **22**(6): 473-491.

Ruger, M. (2003). SuperSwiki - bringing collaboration to the class room. *First Conference on Creating, Connecting and Collaborating through Computing*, Kyoto, Japan, IEEE. 18-21.

Savolainen, R. (2007). Motives for giving information in non-work contexts and the expectations of reciprocity. The case of environmental activists. *Proceedings of the American Society for Information Science and Technology* **44**(1): 1-13.

Schmidt, K. and Wagner, I. (2004). Ordering Systems: Coordinative Practices and Artifacts in

Architectural Design and Planning. *Computer Supported Cooperative Work (CSCW)* **13**(5 - 6): 349-408.

Shannon, C. and Weaver, W. (1949). *The Mathematical Theory of Communications*. Urbana, IL, University of Illinois Press.

Sole, D. and Applegate, L. (2000). Knowledge sharing practices and technology use norms in dispersed development teams. *21st international conference on Information systems*, Brisbane, Queensland, Australia, Association for Information Systems. 581 - 587.

Sonnenwald, D. H. (1995). Contested collaboration: A descriptive model of intergroup communication in information system design. *Information Processing & Management* **31**(6): 859-877.

Sonnenwald, D. H. and Iivonen, M. (1999). An Integrated Human Information Behavior Research Framework for Information Studies. *Library & Information Science Research* **21**(4): 429-457.

Sonnenwald, D. H. and Lievrouw, L. A. (1997). Collaboration during the design process: A case study of communication roles and project performance. *Information Seeking in Context*. V. R. Savolainen and B. Dervin. London, Taylor Graham**:** 179-204.

Sousa, F., Aparicio, M. and Costa, C. J. (2010). Organizational wiki as a knowledge management tool *Proceedings of the 28th ACM International Conference on Design of Communication*, Satildeo Carlos, Satildeo Paulo, Brazil, ACM.

Sousa, F., Aparício, M. and Laureano, R. (2010). Different contributor profiles in an organizational wiki *Proceedings of the Workshop on Open Source and Design of Communication*, Lisbon, Portugal, ACM.

Stettina, C., J. and Heijstek, W. (2011). Necessary and neglected?: an empirical study of internal documentation in agile software development teams. *Proceedings of the 29th ACM international conference on Design of communication (SIGDOC '11)*, ACM. 159-166.

Stocker, A. and Tochtermann, K. (2011). Enterprise Wikis – Types of Use, Benefits and Obstacles: A Multiple-Case Study

Knowledge Discovery, Knowlege Engineering and Knowledge Management. A. Fred, J. L. G. Dietz, K. Liu and J. Filipe, Springer Berlin Heidelberg. **128:** 297-309.

Subrahmanian, E., Monarch, I., Konda, S., Granger, H., Milliken, R., Westerberg, A. and Thendim, g. (2003). Boundary Objects and Prototypes at the Interfaces of Engineering Design. *Computer Supported Cooperative Work (CSCW)* **12**(2): 185-203.

Subrahmanian, E., Monarch, I., Konda, S. L., Westerberg, A. and group, n.-d. (2000). Classification for communication and communication for classification in engineering design. *Worshop on Classification Schemes in Cooperative Work held at CSCW2000*, Philadelphia.

Syan, C. S. and Menon, U. (1994). *Concurrent Engineering: Concepts, implementation and practice*, Chapman & Hall.

Thom-Santelli, J., Cosley, D. R. and Gay, G. (2009). What's mine is mine: territoriality in collaborative authoring *Proceedings of the 27th international conference on Human factors in computing systems*, Boston, MA, USA, ACM.

Trigg, R. H., Blomberg, J. and Suchman, L. (1999). Moving document collections online: the evolution of a shared repository. *Sixth European conference on Computer supported cooperative work*, Copenghagen, Denmark, Kluwer Academic Publishers. 331-350.

Tselios, N., Altanopoulou, P. and Komis, V. (2011). Don't leave me alone: effectiveness of a framed wiki-based learning activity *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, Mountain View, California, ACM.

Tuomi, I. (2000). Data is More than Knowledge: Implications of the Reversed Knowledge Hierarchy for Knowledge Management and Organizational Memory. *Journal of Management Information Systems* **16**(3): 107-121.

Vatrapu, R. K. and Suthers, D. D. (2010). Cultural influences in collaborative information sharing and organization *Proceedings of the 3rd international conference on Intercultural collaboration*, Copenhagen, Denmark, ACM.

Vicente, K. (1999). *Cognitive Work Analysis: Towards Safe, Productive, and Healthy Computer-Based Work*. Mahwah, NJ, Lawrence Erlbaum Associates.

Vidal, R., Mulet, E. and Gómez-Senent, E. (2004). Effectiveness of the means of expression in creative problem-solving in design groups. *Journal of Engineering Design* **15**(3): 285-298.

Viégas, F. B., Wattenberg, M. and Dave, K. (2004). Studying cooperation and conflict between authors with history flow visualizations. *SIGCHI conference on Human factors in computing systems*, Vienna, Austria, ACM. 575-582.

Viégas, F. B., Wattenberg, M., Kriss, J. and Ham, F. v. (2007). Talk Before You Type: Coordination in Wikipedia. *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*. 78-78.

Voida, S. and Greenberg, S. (2009). WikiFolders: augmenting the display of folders to better convey the meaning of files *Proceedings of the 27th international conference on Human factors in computing systems*, Boston, MA, USA, ACM.

Wagner, C. (2004). Wiki: A Technology for Conversational Knowledge Management and Group Collaboration. *Communications of the Association of Information Systems* **13**: 265-289.

Walthall, C. J., Devanathan, S., Kisselburgh, L. G., Ramani, K., Hirleman, E. D. and Yang, M. C. (2011). Evaluating Wikis as a Communicative Medium for Collaboration Within Colocated and Distributed Engineering Design Teams. *Journal of Mechanical Design*

133(7): 071001.

Walz, D. B., Elam, J. J. and Curtis, B. (1993). Inside a software design team: knowledge acquisition, sharing, and integration. *Communications of the ACM* **36**(10): 63-77.

Wang, H.-C., Lu, C.-H., Yang, J.-Y., Hu, H.-W., Chiou, G.-F., Chiang, Y.-T. and Hsu, W.-L. (2005). An empirical exploration of using Wiki in an English as a second language course. 155-157.

Wang, W. Y. and McKeown, K. R. (2010). "Got you!": automatic vandalism detection in Wikipedia with web-based shallow syntactic-semantic modeling *Proceedings of the 23rd International Conference on Computational Linguistics*, Beijing, China, Association for Computational Linguistics.

Wei, C., Maust, B., Barrick, J., Cuddihy, E. and Spyridakis, J. H. (2005). Wikis for supporting distributed collaborative writing. *Society for Technical Communication 52nd Annual Conference*, Seattle, WA.

White, K. F., Gurzick, D. and Lutters, W. G. (2009). Wiki anxiety: impediments to implementing wikis for IT support groups *Proceedings of the Symposium on Computer Human Interaction for the Management of Information Technology*, Baltimore, Maryland, ACM.

Wilson, T. D. (2000). Human Information Behavior. *Informing Science* **3**(2): 49-56.

Wu, D., Yang, D., Koolmanojwong, S. and Boehm, B. W. (2009). Experimental Evaluation of Wiki Technology and the Shaper Role in Rapid Interdisciplinary Requirements Negotiation. *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*. 1-9.

Wu, Q., Irani, D., Pu, C. and Ramaswamy, L. (2010). Elusive vandalism detection in wikipedia: a text stability-based approach *Proceedings of the 19th ACM international conference on Information and knowledge management*, Toronto, ON, Canada, ACM.

Xiao, W., Chi, C. and Yang, M. (2007). On-line collaborative software development via wiki *Proceedings of the 2007 international symposium on Wikis*, Montreal, Quebec, Canada, ACM.

Yamaoka, T., Tsujino, K., Yoshida, T. and Nishida, S. (1998). Supporting Mutual Understanding in Collaborative Design Project. *3rd Asian Pacific Computer and Human Interaction*, IEEE Computer Society. 132.

Yang, D., Wu, D., Koolmanojwong, S., Winsor, B., A. and Boehm, B. W. (2008). WikiWinWin: A Wiki Based System for Collaborative Requirements Negotiation. *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*. 24-24.

Yang, M. (2009). Observations on concept generation and sketching in engineering design. *Research in Engineering Design* **20**(1): 1-11.

Yang, T.-M. and Maxwell, T. A. (2011). Information-sharing in public organizations: A literature review of interpersonal, intra-organizational and inter-organizational success factors. *Government Information Quarterly* **28**(2): 164-175.

Yates, D., Wagner, C. and Majchrzak, A. (2010). Factors affecting shapers of organizational wikis. *J. Am. Soc. Inf. Sci. Technol.* **61**(3): 543-554.

Yitzhaki, M. and Hammerschlag, G. (2004). Accessibility and use of information sources amoung computer scientists and software engineers in Israel: Academy versus industry. *Journal of the American Society for Information Science and Technology* **55**(9): 832-842.

Zhao, X., Xiao, W., Chi, C. and Yang, M. (2011). Integrating Twitter into Wiki to support informal awareness *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, Hangzhou, China, ACM.

APPENDICES

Appendix A: Interview protocol

1. **Briefly explain the study goal and the objectives of today's interview.**
   "The goal of this study is to understand better the ways in which design team members share information during the design process. For today's interview, the objective are to study what information and how individual members share with others, and how individual members use the project wikis."

2. **Explain the interview procedure.**
   "*I will ask you a series of questions regarding your tasks, information sharing practices, and experience using the project wiki. The interview will take approximately 45 minutes. You do not have to answer any questions that you feel uncomfortable, and you can stop the interview at any point."*

3. **Have participant sign consent form.**
   "*Before we begin, I would like you to read and sign a consent form*."
   Give the participant a consent form and give 2-3 minutes to read and sign.
   "*Please let me know if there is any question about the consent form.*"

4. **Begin interview.**
   Backgrounds
   1. What is your educational background?
   2. Have you taken job-related training?
   3. What is your area of expertise?
   4. How long have you worked at (the organization name)?

   Project overview (Project manager only)
   5. Please describe the project (goals, expected outcomes, and team structure)
   6. Please describe the project wiki
   7. What was your expectation on how the project wiki to be used by the team members?

   Roles & tasks
   8. What is your role in [the project name]?
   9. What are you working over the past few weeks in this project?
   10. How would you describe your tasks you typically do in this project?

   Information sharing
   11. What information do you need from other team members in order to work on your tasks? (Follow-up: What documents did you need from…?)
   12. What information do you need to share or exchange with other team members? (Follow-up: What documents did you shared with others…?)
   13. How do you share or exchange [the information mentioned in Q2] with others?
       a. Method or technique

    b. To whom do you share
    c. Reason for sharing
14. How did you think the project wiki would help information sharing in this project?

Wiki

15. In general, how do you use the project wiki?
    a. How often do you access the project wiki?
    b. Why do you access the project wiki?
16. Please describe how you use the project wiki to support your work/tasks
17. What information have you put on the project wiki?
18. How do you make a decision what to post on the wiki?
19. What happened with the information you shared with others?
20. What do you like about the wiki?
21. What do you dislike about the wiki?
22. Overall, how do you describe your experience using the project wiki?
23. Critical incident question
    a. Please describe an event when you use the project wiki that you feel having a strong influence on the project.
    b. What led to the event?
    c. What did you actually do?
    d. How did your action influence the project?
24. Is there anything else you would like to add?

5. Thank you a participant for his/her participation.

Appendix B: Code development/translation

1. CWA dimensions

Work environment

- Organization (size, industry, and organizational values)
- Wiki adoption

Work-domain

- Project (size, duration, and goal)

Organizational and social structure

- Project structure (roles and responsibilities)
- Collaboration

Activity

- Information sharing (goals, constraints, activities, technologies and tools)
- Wiki task (goals, constraints, activities, technologies and tools)

User's resources and values

- Experience with software design
- Experience with wiki
- Perception
- Attitude

2. Means-ends structure dimensions

Project

- Goals – Why does the project exist? What does the project try to achieve?
- Constraints – What are some limitations of how the project could be operated?
- Priorities – What do the project members need to do in order to accomplish the goals with given constraints?
- Functions – What are general functions the project members perform to meet its priorities?

- Work processes – How do the project members physically carry out the project functions?
- Resources – What are tools and technologies the project members use to perform their work processes?

Project wiki use

- Goals - Why do the project members use the project wiki?
- Constraints – What limitations do the project members have when using the project wiki?
- Priorities – what do the project members want to achieve when using the project wiki?
- Functions – what are general functions of the project wiki?
- Work processes – What process do the project members physically perform when sharing information through the project wiki?
- Resources – what features and other tools do the project members use when sharing information through the project wiki?

Appendix C: Projects' means-ends analysis

1. Project A: Web-based workspace

Goals **Constraints**	• to develop a web-based application to meet basic campus needs for courseware, and support group collaboration • Software design process the project team follows • Organizational factors including long-term goal, current products and services, and political forces • Project schedule • Stakeholders including the director, sponsors, users, and working group committee • Project scope • Social conventions in the group • External collaboration • Project members' skills and experiences
Priorities	• Designing and developing a usable web application that users like • Delivering the application on time or with reasonable delays • Following the agile and getting real principles • Working successfully as a team • Collaborating effectively with each other
Functions	• Planning • Management • Requirements • Design • Development • Testing • Release
Work Processes	• Defining the project's goals • Creating a project definition document • Designing user research • Reviewing user input related to the project from Firebox • Reviewing user input related to the project from RT • Conducting literature review • Conducting interviews of users • Conducting focus group interviews • Creating a report from user research • Presenting user research findings to the team • Identifying requirements

	• Drafting functional requirements • Prioritizing requirements • Writing a one-pager/mantra • Discussing and finalizing functional requirements • Doing a work breakdown (Post-it) and estimation • Developing use-case scenarios for students, staff, and instructors • Creating meeting agenda • Scheduling a meeting • Facilitating a meeting • Creating an initial project schedule • Adjusting project timeline throughout the project life cycle • Updating the project schedule • Creating task flow diagrams • Creating screen flow diagrams • Sketching initial design ideas • Sketching wireframes on paper • Documenting rationale • Making high-level mockup • Drafting a design specification document • Reviewing the design specification • Developing a prototype • Developing template (front-end development) • Initially editing text in the user interface • Recruiting and scheduling study participants • Creating a project wiki site • Creating a project folder in a file server • Taking meeting notes • Reviewing meeting notes • Designing a usability study • Recruiting participants for a usability study • Scheduling participants • Creating a usability schedule on the project wiki • Preparing materials for the usability study • Conducting a usability study • Taking notes during a usability study • Debriefing after each usability study • Typing up notes from usability study • Analyzing usability study data • Compiling usability findings • Presenting usability study results to the team • Generating design recommendations from usability findings • Revising design based on the usability findings • Reviewing the design specification • Updating the design specification

	• Revising the prototype • Creating an application back-end model • Documenting APIs • Testing APIs • Writing codes • Working on visual design • Attending regular project meetings (3 times a week during the beginning of the project, and once a week toward the end) • Having informal face-to-face meetings • Writing test plans • Having undergraduate student staff test • Reporting bugs • Assigning bugs • Fixing bugs • Monitoring bugs • Retesting the application • Having a professional staff testing day • Retesting the application after the professional staff testing • Reviewing accessibility • Text editing • Identifying different types of documentation for the application • Writing the documentation for end users • Updating Web site content • Coordinating the release • Notifying users of the release • Creating a release checklist • Preparing marketing information • Announcing the release • Creating a mailman list • Communicating with higher level management • Coordinating with team leads • Forming the project team • Exploring the application • Participating in group decision-making • Planning testing schedule
Physical Objects	• Technology: wiki, email, a shared file server (Satchmo), ShareSpaces, BackpackIt, Illustrator, Photoshop, MS Word, MS Excel, MS visio, MS project, IRC, Bugzilla, RT system, whiteboard, post-it notes • Information items: design specifications, functional requirements, usability study findings, meeting notes, mock ups, wireframes, persona, task flows, schedule.

2. Project B: Online discussion board enhancement

Goals **Constraints**	• To add more innovative, advanced, and administrative features to an online discussion board application • To release features in multiple phases • Schedule • Pre-identified user needs and existing functional requirements • Technical considerations • Existing functionality of the current application • Project scope • Political forces from the university • Project members' skills and experiences • Project members' availability and workload outside of the project
Priorities	• Incorporating user feedbacks gathered through different channels • Building the application on the Solstice application framework • Aiming for less paper • Following a user-centered design process to design and develop the application • Completing the project on schedule or with reasonable delays • Keeping the project goals in line and making sure that we either consciously sticking to them or consciously choosing for good reasons to go out of the original plan • Meeting performance criteria • Collaborating and supporting each other toward the success of the project.
Functions	• Planning • Management • Requirements • Design • Development • Testing • Release
Work Processes	• Facilitating meetings • Creating a project wiki site • Scheduling weekly meetings • Attending weekly meetings • Creating functional requirements • Designing a survey • Conducting a survey research • Analyzing survey results

	• Making recommendations for new features based on the survey results • Reviewing design • Providing feedback and suggesting improvements to the design team • Tracking work progress • Coordinating a staff testing day • Creating bug reports • Assigning bugs • Fixing bugs • Monitoring work progress • Adjusting the project timeline • Coordinating the release • Writing test plans • Creating the next push checklist on the project wiki • Developing Codes • Sketching on a whiteboard • Creating task flow diagrams • Reviewing user feedback from Firebox and RT • Conducting literature review • Reviewing log data of the current discussion board application • Sketching wireframes on paper • Documenting design options • Documenting design decisions and design rationale • Making high-level mockup • Drafting a design specification document • Reviewing the design specification • Designing a quick usability study • Finding participants • Scheduling participants • Conducting a usability study • Taking notes from the usability study • Analyzing data from the usability study • Compiling usability study results • Recording usability study results • Sharing usability study results • Generating design recommendations based on the usability study results • Presenting results and recommendations to the team • Working template (front-end development) • Writing test plans • Testing with undergraduate student staff • Fixing bugs • Addressing UI bugs • Testing bug fixes • Coordinating a professional staff testing day

	• Editing text in the UI • Editing help documentation • Updating Web site content • Notifying users of the release • Creating a release checklist • Preparing marketing information • Announcing the release • Revising workshop content
Physical Objects	• Technology: wiki, email, a shared file server (Satchmo), ShareSpaces, BackpackIt, Illustrator, Photoshop, MS office applications, IRC (project channel), Bugzilla, Request tracking system, Gobby,w hiteboard, post-it notes • Information items: Functional requirements, task flow diagrams, screen flow diagram, usability findings, design specifications, project schedule, meeting notes, UI template to-do list, release checklist, test plans, wireframes

3. Project C: Ticket Reprice and Reissue System

Goals **Constraints**	• to develop a web-based application to support self-service ticket reissuing to save cost and reduce the workload of the reservation office • Agile software development process the project team follows • Organizational factors including long-term goal, current products and services, and political forces • Project schedule • Project scope • Social conventions in the group • External collaboration with a vendor and IT development team • Project members' skills and experiences • Existing company website
Priorities	• Designing and developing software that meets the requirements and usable for customers • Delivering the application on time within estimated budget • Collaborating and communicating effectively with stakeholders
Functions	• Planning • Design • Development • Testing • Release • Communication

Work Processes	• Managing team members, forming the project team - And so we added them to - we added at least their team names to the wiki • Running beta testing - And then, so we've got alpha testing, which is integration alpha, and then beta testing. • Keeping track of work - they had a burn-down chart of the effort they put in • Planning - Well, I would say the QA team as a whole and then myself just from an overview to say, okay, I agree that you guys - it looks like you put in the effort; you've tested it. Their manager, the QA manager probably spent some time looking at it • Releasing the code - in August, it'll be a year since we deployed the code. • Creating design mockups • Revising design mockups • Creating use cases • Developing test scripts • Reviewing use cases • Revising use cases • Creating the project wiki • Debriefing lessons learned at the end of each release • Coding • Creating a bug report • Adding a bug to a bug issue log • Assigning a bug to appropriate persons • Reporting status to the stakeholders • Defining the project scope • Estimating a project timeline and budget • Allocating resources • Planning tasks • Coordinating with the vendor • Communicating with the steering committee • Triaging bugs • Having a daily meeting • Planning the development cycles • Setting up a meeting • Designing an interaction flow • Planning usability studies • Conducting usability studies

Physical Objects	• Technology: Mantis (bug tracking tool), task manager tool, wiki, email, powerpoint, photoshop • Information items: use cases, burn-down chart, test scripts, design mockups, blue print, bug reports, a bug issue log, weekly status report, calendar, vendor's bug resolution, meeting notes, back-end processing type documentation

4. Project D: Virtual Assistant

Goals **Constraints**	• to develop an online, voice-enabled virtual assistant to improve customer self-service on the company website • Agile software development process the project team follows (1-4 releases a year) • Organizational factors including long-term goal, current products and services, and culture of innovation • Project schedule • Stakeholders including steering committee, sponsor, IT development team for backend processing, and vendors • Project scope – innovation project • Social conventions in the group • External collaboration – needing to collaborate with other projects in the company that could affect the VA functions. • Project members' skills, experiences and availability • Existing company website
Priorities	• Designing and developing software that meets the requirements and usable for customers • Delivering the application on time within estimated budget • Collaborating effectively with each other
Functions	• Planning • Design • Development • Testing • Release • Communication
Work Processes	• Defining the project scope • Selling the project to the organization • Administrating the project progress • Ensuring that the product and project match with the customer needs • Facilitating • Updating/reporting work progress • Communicating between the marketing business group and the development group

	• Setting up the project wiki • Writing a project charter • Documenting the business rules • Overseeing the company website • Organizing the project wiki • Developing a project schedule • Updating requirements • Notifying changes to the company website • Notifying a new feature to be added to the VA • Adding test questions to improve the VA • Releasing the VA to the public • Planning the software release • Communicating with executives • Testing the VA
Physical Objects	• Technology: Projector, Chat log, Excel, FirstClass (email client) • Information items: meeting notes, schedule, business rules, audio scripts, test plan (spreadsheets)

5. Project E: Customer facing web

Goals **Constraints**	• To improve customer experience on the company website • Agile software development process the project team follows • Organizational factors including long-term goal, current products and services, and political forces • Project schedule • Stakeholders including director • Project scope • Social conventions in the group • External collaboration (4-5 different teams working together) • Project members' skills, experiences and availability Existing company website and multiple systems • Management styles
Priorities	• Designing and developing software that meets the requirements and usable for customers • Delivering the application on time within estimated budget • Collaborating effectively with each other
Functions	• Planning • Design • Development • Testing

	• Release • Communication
Work Processes	• Developing a work backlog • Implementing features listed in a backlog of work • Prioritizing the task list • Developing features • Fixing bugs • Planning sprint work (weekly plan) • Testing the code • Developing test cases • Adding test cases to the wiki • Reviewing test cases • Editing test cases in the wiki • Creating ScrumWorks stories • Having daily scrum meetings • Creating a work backlog • Prioritizing tasks • Release planning
Physical Objects	• Technology: Projector, ScrumWorks, Wiki, SharePoint • Information items: test cases, scrum backlog, stories

Appendix D: Reviewed artifacts from the project wiki sites

Artifact title	Format
Project A wiki site	
A list of competitor tools	Links
A list of Dev-fix bug reports	Links
A list of UI consideration bug reports	Links
Action items	Wiki page
Adoption ideas	Wiki page
Design specifications	Wiki pages
Functional requirements	Files
Functional requirements feedback	Wiki page
Issues to discuss later	Wiki page
Literature review	Files
Needed functionality (pre-bug fixes)	Wiki page
Prioritized user needs	Wiki page
Project debrief meeting notes	Wiki page
Project definition	Wiki page
Project schedule	Image
Research data	Links
Research findings	Files
Research protocol	Wiki page
Scenarios	Wiki pages
Screen flow diagram	Image
Task flow diagram	Image
Template to-do list	Wiki page
Tool vision (One-pager & mantra)	Wiki page
Usability study findings	Wiki page
Usability study notes	Wiki page
Usability study page	Wiki page
Usability study protocol	Wiki page
Usability study schedule	Wiki page
Workflow problem	File
Working prototype	Link
Project B wiki site	
Old tool phase out plan	Wiki page
Tool outage notice	Wiki page
Notice to tool users	Wiki page
Feature list from past project	Wiki page
Project schedule	Wiki page
Literature review	Links
Application log data	Files
User feedback and requests	Files

Artifact title	Format
Research protocol	Wiki page
Research findings	Wiki page
Quick usability study	Wiki page
Scenario	Wiki page
Functional requirements	Wiki page
Task flow	Images
Note about banning feature	Wiki page
Migration plan	Wiki page
Design specifications	Wiki pages
A list of UI consideration bug reports	Link
Working prototype	Link
A list of Dev-fix bug reports	Link
Website help documentation	Wiki page
Ideas for current and future development	Wiki page
Meeting notes	Wiki page
Project C wiki site	
Project charter	File
Task manager	File
Project calendar	File
Presentation to executive leadership	File
Project Presentation	File
Feature guide	File
Questions for vendor	File
Bugs	Link
Issue and business rule log	File
Mock ups	Images
Usability tests	Wiki page
Blueprint	File
Gap analysis	File
Use cases	Wiki page
Fare discussions	Wiki page
Test scenarios	File
Test scenario failures	File
Validation script	File
Master list of test scenarios	File
Screen flow	File
Errors screenshots	File
Changes tracking log	File
Log files directory	Wiki page
Project D wiki site	
Project overview	Wiki page
Weekly goals	Wiki page
Project status	File
Task log tracking	File

Artifact title	Format
Completed tasks	File
Issues log tracking	File
Risk log tracking	File
Weekly status meeting minutes	File
Persona	File
First design iteration	Wiki page
First usability test	Wiki page
Usability share	File
Main style guide	Wiki page
Technical notes	Wiki page
Roles and responsibilities	Wiki page
Technical infrastructure	File
Technical documents	File
Navigation and error messaging	File
Survey results	File
Charter V.3 – approved & update	File
Initial project plan	File
Project E wiki site	
Project status	Wiki page
Project backlog	Wiki page
User testing bugs and triage	Wiki page
Use case	File

Appendix E: Participants' profiles

Participants	Age	Gender	Educational background	Working period	Technical skills	Prior Wiki experience
Project A						
Project manager1	36-45	M	Business	6 months	Medium	Low
Researcher 1	36-45	F	Psych Education	4 years	Low	None
Researcher 2	26-35	F	English Studies	5 years	Low	Low
Designer 1	26-35	M	HCI	2 years	Medium	High
Designer 2	26-35	M	Technical Comm	4 years	Medium	High
Designer 3 (myself)	26-35	F	Info Science	3 years	Medium	Medium
Software developer 1	18-25	M	Informatics	4 years	High	High
Software developer 2	36-45	M	Com Science	7 years	High	Medium
QA engineer	18-25	M	Informatics	2 years	High	Medium
Project B						
Director	36-45	M	Political Science	8 years	Medium	Medium
Project manager	36-45	F	Political Science	4.5 years	Medium	Medium
Designer 1	18-25	M	Informatics	4.5 years	High	High
Designer 2	26-35	F	HCI	9 months	High	Low
Software developer 1	18-25	M	Mathematics	2 years	Medium	Medium
Software developer 2	26-35	M	Com Science	4 years	High	High
QA engineer	18-25	M	Informatics	2 years	High	Medium
Project C						
Project manager	36-45	F	Marketing	22 years	High	Low
Interaction designer	18-25	F	Industrial Engineering	4 years	Medium	Low
QA system analyst	26-35	F	Com Science	5 years	High	Low
Pricing analyst	26-35	M	Com Science, MBA	3 years	Medium	Medium
Project D						
Project manager	36-45	F	Elementary education, MIS	12 years	High	Medium
Product manager	36-45	F	English	2 years	Medium	Low
E-commerce	26-35	M	Broadcast	5 months	Medium	Low
Customer care lead	36-45	F	Transportation and marketing	20 years	Low	Low
Customer care 1	18-25	F	Interior design	4 months	Low	Low
Customer care 2	26-35	F	English	3 years	Low	Low
Customer-facing dev	36-45	M	Com Science	10 years	High	Medium
Project E						
Program manager	36-45	M	Business	1 year	High	Medium
Developer manager	26-35	M	Com Science	2 years	High	High
Developer	18-25	M	Com Science	10 months	High	Low
QA engineer	18-25	M	Industrial Tech	1 year	High	Low
QA engineer manager	26-35	M	Com Science	9.5 years	High	High
Network engineer	26-35	M	Business, Com Science	6 years	High	Medium

Appendix F: Consent form

UNIVERSITY OF WASHINGTON
CONSENT FORM
User of Wikis for Information Sharing in Design Teams (Phase 1)

Researcher: Ammy Jiranida Phuwanartnurak, Ph.D. student, Information School, University of Washington, Seattle, WA 98195-2840. Ph. 206-221-6402. jiranida@u.washington.edu (I cannot ensure the confidentiality of information sent via e-mail.)

Researchers' statement

I am asking you to participate in a research study. The purpose of this consent form is to give you the information you will need to help you decide whether to be in the study or not. Please read the form carefully. You may ask questions about the purpose of the research, what I would ask you to do, the possible risks and benefits, your rights as a volunteer, and anything else about the research or this form that is not clear. When I have answered all your questions, you can decide if you want to be in the study or not. This process is called 'informed consent.' I will give you a copy of this form for your records.

PURPOSE OF STUDY

The goal of this research is to better understand how interdisciplinary design teams use wikis to collaborate on a design project. I will observe design teams in the design process, and interview design members.

PROCEDURES

If both you and your team agree to participate in this study, I will ask you to take part in a 45-minute interview. In the interview, I will ask questions regarding your information sharing practices and use of a project wiki. For example, "What information do you share with your team members?", "How do you share and exchange information with others?", and "How often do you access a project wiki?" You do not have to answer every question. All interviews will be scheduled in advance at the participant's convenience.

With your permission, I would like to audiotape the interviews so that I can have an accurate record. Only I will have access to the audiotapes, which will be kept in a locked file cabinet. I will transcribe your interview tape within 2 weeks, assign a study code to the transcript, and destroy the tape within four years of your interview. Please indicate below whether you give your permission for the interview to be audiotaped.

RISKS, STRESS, OR DISCOMFORT

Some people feel that taking part in research is an invasion of privacy. Some people feel self-conscious when they are observed. I have addressed concerns about your privacy in the following section of this consent form.

BENEFITS OF THE STUDY

I hope that the results of this study will be used to develop tools to improve the way information is shared among design team members. You may not directly benefit from taking part in this study.

OTHER INFORMATION

Taking part in this study is voluntary. You can stop at any time. Information about you is confidential. I will code the study information. I will keep the link between your name and the code in a separate, secured location until April 2011. Then I will destroy the link. If the results of this study are published or presented, I will not use your name. The following groups may need to review study records about you: Institutional oversight review offices at the UW; and federal regulators.

I may want to re-contact you to clarify information from your interview. IN that case, I will telephone you and ask you for a convenient time to ask you additional questions closely related to your interview. Please indicate below whether or not you give your permission for me to re-contact you for that purpose. Giving your permission for me to re-contact you does not obligate you in any way.

____

____

Printed name of researcher   Signature of researcher
Date

<u>Subject's statement</u>

This study has been explained to me. We volunteer to take part in this research. I have had a chance to ask questions. If I have questions later on about the research I can ask one of the investigators listed above. If I have questions about my rights as a research subject, I can call the University of Washington Human Subjects Division at (206) 543-0098. I give the researchers permission to observe and interview me as described in this consent form. I will receive a copy of this consent form.

_____ I give my permission for the researcher to audiotape my interview.
_____ I do NOT give my permission for the researcher to audiotape my interview.
_____ I give my permission for the researcher to re-contact me in order to clarify information from my interview.

_____ I do NOT give my permission for the researcher to re-contact me.

____

Signature of subject   Printed name   Date