

# Effective Communication, Collaboration, and Coordination in eXtreme Programming: Human-Centric Perspective in a Small Organization

**Deepti Mishra**

*Department of Computer Engineering, Atilim University, Ankara, Turkey*

**Alok Mishra**

*Department of Software Engineering, Atilim University, Ankara, Turkey*

## ABSTRACT

Effective communication, collaboration, and coordination are important contributing factors in achieving success in agile software development projects. The significance of the workplace environment and tools are immense in effective communication, collaboration, and coordination among people performing software development. In this article, we study how the workplace environment and the effective use of tools like whiteboards, status boards, and so forth for exchanging information improved communication, collaboration, and coordination without compromising the ability to do individual work by developers in a small-scale software development organization. Based on experience and an extensive literature review of communication, collaboration, coordination, and the significance of these in the workplace environment, a survey questionnaire was developed to collect data and observe the effect of these in a small software development organization. Our study indicated appropriate workspace environment has a positive effect on communication, collaboration, and coordination in small organizations developing software using eXtreme Programming (XP). © 2009 Wiley Periodicals, Inc.

## 1. INTRODUCTION

Effective communication, collaboration, and coordination are the main contributing factors for success in agile methods. Communication is defined as imparting or interchanging thoughts, opinions, or information by speech, writing, or signs. Collaboration is defined as working together to accomplish a task and discussing with each other to solve difficult problems. Effective collaboration includes both individually focused tasks and interactive group work. Coordination is defined as the harmonious adjustment or interaction of different people or things to achieve a goal or effect.

Learning is not possible without communication (Lei, Slocum, & Pitts, 1999) as communication is a prime factor for learning. Communication can be verbal or nonverbal through media like books, white papers, and pictures or through observations. Without information exchange, employees cannot learn from other experts and the process of knowledge

Correspondence to: Deepti Mishra, Department of Computer Engineering, Atilim University, Ankara 06836, Turkey. E-mail: [deepti@atilim.edu.tr](mailto:deepti@atilim.edu.tr)

acquisition and sharing would be inhibited (Ribbens, 1997; Schein, 1993). Duncan and Moriarty (1998) describe communication as the human activity that links people together and creates relationships. It is at the heart of meaning-making activities.

There are various channels of communication, but face-to-face communication is found to be the most effective as it provides instant feedback and multiple cues like expression, emotions, and personal focus. The knowledge acquired through face-to-face communication can be retained for a limited time, then it starts diminishing gradually. So, some tools like papers, whiteboards, and so forth, may be used to store information intended for future use. These tools are also useful to access information about the project when many people are working together on one project or when multiple teams (consisting of many people) are simultaneously working on different parts of the same project and coordination among them is important.

Small teams are more effective in coordination, communication, and collaboration than are large teams (Cockburn, 2000; Curtis, Curtis, & Iscoe, 1988). Small teams with two to ten people are known to perform better than the same individuals working alone, especially while dealing with complex problems. This view is also supported by Cockburn (2000). According to Cao and Balasubramaniam (2007), agile development involves intensive teamwork and high task interdependence. As task interdependence increases, the need for coordination becomes greater (Cao & Balasubramaniam, 2007).

The physical design of the workspace also plays an important role in effective communication and collaboration. If people working on the same team are close to each other physically then it may be easier for them to communicate and collaborate effectively when required to eliminate ambiguity. Furthermore, workspace design will be effective if it facilitates communication and collaboration and also provides an environment to do individual work effectively. There are few studies done so far that observed the importance of physical design of workspace for effective communication and collaboration, which make up the backbone of software development, especially using agile methods (Beck, 2000; Cockburn, 2000; Heerwagen, Kampschroer, Powell, & Loftness, 2004; Sharp & Robinson, 2003). According to Heerwagen et al. (2004), collaborative work environment requires spaces, furnishings, and technologies that support both individual focus and group interaction, while also facilitating transitions between these activities.

In this article, we have studied the physical settings of a small-scale software development organization working on a large, complex project using eXtreme Programming (XP) for a domain in which they had no previous experience. XP is one of the approaches of agile methods. we examined how different people (developers, business experts, customers, etc.) working on this project communicated and collaborated effectively without disrupting individually focused tasks for successful completion of the project. There are two small teams working simultaneously on different parts of the product (project). This study shows how the communication with customers, within a team and across two teams, took place during the software project using agile methods. Furthermore, we learned how the physical design of the workspace (proximity of different rooms on the same floor, arrangement of furniture, presence of whiteboards, communal area, etc.) enhanced effective communication and collaboration without compromising the ability to work alone in an effective way. This study is based on experiences; an extensive literature review of communication, collaboration, and coordination; and significance of these factors in the workplace. A survey questionnaire was developed to collect data and observe the effect of physical environment and tools on communication, collaboration, and coordination in a small organization using agile methods. This study will provide a guideline for prospective office designers who are adopting agile methods.

The article is organized in the following manner: Section 2 presents a comprehensive literature review related to communication, collaboration, and coordination significance in agile methods. Tools like whiteboards and status boards and office layout in an organization can enhance productivity and quality of software. In Section 3, we illustrate the case study experience within agile software development. Section 4 reports the research method, discussion, and recommendations based on data collected through a questionnaire. Section 5 summarizes the conclusion and directions toward further research in this area.

## **2. LITERATURE REVIEW**

### **2.1. Communication, Collaboration, and Coordination as Key Enablers for Project Success**

Software development is a cooperative game of invention and communication. There is nothing in the game but people's ideas and the communication of those ideas to colleagues and to the computer (Cockburn, 2000). In complex situations, communication effectiveness is particularly critical to project success where multiple and integrated stakeholders teams are involved and where "time to market" and project efficiency are key drivers (Elliott, 2000). This is supported by many researchers in their studies (Beck, 2000; Bostrom and Thomas, 1983; Edstrom, 1997). Problems with projects can invariably be traced back to somebody not talking to somebody else about something important (Beck, 2000). The more you communicate, the clearer you can see exactly what needs to be done and the more confidence you have about what really does not need to be done (Beck, 2000). Focusing on skills, communication, and community allows the project to be more effective and more agile than focusing on processes does. People's individual skills and collaboration, conversations, and communications between them enhance flexibility and innovation (Highsmith, 2002). According to Heerwagen, Kampschroer, Powell, and Loftness (2004), knowledge work is both highly cognitive and highly social. Workers need time alone to think and develop ideas, drawing on their own memory, insight, and analytical skills. Yet, for ideas and concepts to become useful to an organization, they must be available to others for scrutiny and further development (Heerwagen et al., 2004).

### **2.2. Face-to-Face Is the Most Effective Means of Communication**

Software development is a cooperative game of communication, implying that a project's rate of progress is linked to how long it takes information to get from the head of one person to the head of another (Cockburn, 2000). Media Richness Theory (Daft & Lengel, 1986; Daft, Lengel, & Trevino, 1987) suggests that the richest communication channel enables instant feedback, is capable of transmitting multiple cues, has a high language variety, and can transmit emotional contents. Face-to-face is the richest form of communication because it is able to transmit multiple cues (e.g., voice inflection, body language) and to facilitate shared meaning with rapid mutual feedback and personal focus, feelings, and emotions infusing the conversation (Daft, Lengel, & Trevino, 1987). From a functional perspective, informal face-to-face interactions aid understanding and problem solving due to the enriched context, including facial expressions, gestures, posture, appearance, and reactions of other people (Kendon, 1990). Face-to-face interaction is also more flexible and can respond better to ambiguity and uncertainty (Allen, 1971). It is further supported by Cockburn (2000) that the most effective form of communication (for transmitting ideas) is interactive and face-to-face.

Also, Korkala, Abrahamsson, and Kyllonen (2006) described that face-to-face communication is identified as the most efficient means of communication between participants. Moreover, the daily collaborative work of business people and developers demands efficient verbal communication between the customer and developers (Korkala, Abrahamsson, & Kyllonen, 2006). Based on the existing literature, Kraut and Streeter (1995) found that formal communication (e.g., structured meetings, specifications, inspections) is useful for routine coordination whereas informal communication (e.g., hallway conversation, telephone calls, workshops) is needed in the face of uncertainty and unanticipated problems, which are typical of software development. McLuhan (1964) has used the terms “hot” and “cool” when discussing different media for communication. Ambler (2002) and Cockburn (2002) have also adapted the use of “hot” and “cold” communication media. They argue that “hot” communication channels provide more information than “cold” media. According to Ambler (2002), face-to-face communication is the “hottest” communication channel, whereas e-mail is positioned as a “colder” communication channel. The use of “cold” communication media can either complement “hot” communication media or be used only in the absence of “hot” communication media. Face-to-face communication is not free from pitfalls, however.

### 2.3. Role of Whiteboards and Status Boards in Exchange of Knowledge

According to Finsterwalder (2001), verbal communication is not always well thought out and the contents of the conversation can be forgotten in time. Therefore, it is effective to use face-to-face communication for acquiring knowledge and clarifying issues related to the current task. If the information gathered now is going to be used in the future, it should be stored somewhere; that is where artifacts and tools like whiteboards, electronic display, flipcharts, and so forth can be used. Also, when numbers of people are working as a team and/or many teams concurrently working on different parts of a product, they must have continual access to the latest information related to the project for effective coordination. Although verbal interaction is the primary channel for communication (Spencer, Coiera, & Logan, 2004), it is only one of a number of channels used for communication. Also, unnecessary verbal interaction can result in loss of important time of the persons who are interacting and may create disturbances and distractions for other people present in the same room. Overhearing people talk especially in a shared context, triggers an involuntary reflexive response that switches attention from the task to the talking individuals (Pashler, Johnson, & Ruthruff, 2001).

Sometimes when the requested information does not require further analysis (e.g., facts such as status of the project), it can be acquired effectively (without using verbal communication) with the use of information-rich tools like whiteboards, electronic displays, flipcharts, and so forth. Traditionally handwritten whiteboards (Wears, Perry, Wilson, Galliers, & Fone, 2007), electronic displays, and other such tools play a significant role in inter- and intra-group communication and coordination. Important information can be made visible or available to everyone with the help of these tools. Their contribution in other fields such as hospitals/trauma centers is stated in a number of studies (Boger, 2003; France, Levin, Hemphill et al., 2005; Jensen, 2004; Marinakis and Zwemer, 2003; Sinnott, 1990; Xiao, Schenkel, Faraj, Mackenzie, & Moss, 2007) in which communication and coordination between different medical staff is of utmost importance.

## 2.4. Importance of Communication, Collaboration, and Coordination in Agile Methods

Agile software development approaches tend to rely on effective personal communication and collaboration between the project participants (Abrahamsson, Salo, Ronkainen, & Warsta, 2002; Cockburn, 2002). Collaboration among participants has two major forms: One is concrete working together to accomplish a task, and the other is discussing with each other to solve some difficult problems (Wu & Sahraoui, 2005). Agile methods work with volatile requirements and embrace personal communication between the participants (Abrahamsson et al., 2002). Because the level of ambiguity can be considered high in the form of unstable requirements, it seems that personal communication is a natural choice for agile development (Korkala, Abrahamsson, and Kyllonen, 2006). Melnik and Maurer (2004) discussed the role of conversation and social interactions as the key elements of effective knowledge sharing in an agile process. Knowledge is considered to be socially constructed and collectively held. They concluded that the higher the level of abstractionism (complexity), the more is the need for interactive knowledge sharing via direct verbal communication (Melnik & Maurer, 2004).

## 2.5. Factors that Affect Software Quality and Productivity

The issue of software productivity and software quality haunts us perpetually with no single solution (Jain & Ting, 1989). Ambler (2005) considers agile quality to be a result of practices such as effective collaborative work, incremental development, and iterative development as implemented through techniques such as refactoring, test-driven development, modeling, and effective communication techniques. Walston and Felix (1977), Boehm (1987), and Vosburgh, Curtis, Wolverton et al. (1984) found that the effect of tools is relatively small, whereas the impact of people and organization is significant on software productivity. Curtis, Curtis, and Iscoe (1988) have studied software development using a layered behavioral model and have concluded that the design of software systems must be treated as a learning, communication, and negotiating process. The common denominator of these three parts is a small team. They also found that design collaboration is more effective when small teams are formed. Additionally, this collaborative problem solving is related to productivity. The communication remains manageable in small teams, and small teams suffer less from conflicting views. A quality product also results when the application knowledge is widely spread across the team, and with smaller teams it is easier to propagate this knowledge (Jain & Ting, 1989). It has been shown that the use of agile methods can be beneficial for product manageability, visibility, and team communication (Larman, 2003) as well as ensuring frequent feedback from the customer (Rising & Janoff, 2000). Rico (2008) showed through his study that agile methods are linked to benefits such as improvement in cost efficiency, productivity, quality, cycle time, and customer satisfaction. At Motorola, a selected set of XP practices was used in the field of safety critical systems (Grenning, 2001). In that case, the use of XP practices was reported to have 53% improved average quality compared to plan-driven software development process.

## 2.6. Importance of Office Layout in Effective Communication

Office layout or physical space also plays an important role in effective communication. If all team members are physically scattered or if the business expert is not available

near development teams then communication will suffer and that will affect quality and productivity of the product. There have been few studies so far to observe the importance of physical space to effective communication in software development organizations, which is the backbone of software development especially using agile methods. Cockburn (2000) described three separate effects that office layout has on communication costs within a project:

- The reduction in cost when people discover information in background sounds (osmotic communication: indirect information transfer through overhearing conversations or simply noticing things happening around you)
- The overall cost of detecting and transferring information (erg-seconds; the term *erg-seconds* captures the cost to transmit a discrete piece of information to other members of a team in terms of both labor or effort and time)
- The lost opportunity cost of not asking questions

The three magnify the effects of distance in office seating. People sitting close to each other benefit in all three effects; people sitting in separated locations suffer in all three (Cockburn, 2000). Deliberate movement to engage someone in conversation drops off dramatically after 30 meters (Heerwagen et al., 2004). Separation by more than 30 meters is equivalent to being in different buildings, if not in different geographical locations (Allen, 1971). Sharp and Robinson (2003) described the physical setting of a project using extreme programming. Beck (2000) described the ideal work space for XP is to have one big room with little cubbies around the outside and powerful machines on tables in the middle. He also stressed that different teams should be separated from each other. Cubicle walls should be of half heights or eliminated entirely (Beck, 2000). A common workplace solution to enhance collaboration is the provision of informal group spaces adjacent to or interspersed among personal workstations. The spaces have moveable furnishings, are located in the open, and are often shared by different work groups. The intent of the space is to support spontaneous meetings and informal work (Heerwagen et al., 2004).

### 3. CASE STUDY

The working environment is a key point in agile methods to increase communication and interactions. In this section, we describe physical space where software development was performed and how the physical space affected the communication between team members, improved the quality of development, and how they fit with used development methods, especially agile software development approaches.

Ideally agile methods (especially XP) advocate the use of an open plan office where everyone (including developers, business experts, etc.) sits and interacts. This design was not possible in this case as development teams were given a small office with three rooms on the same floor. There were two teams simultaneously working on the project to launch in the market. As it was a market-driven software development, there were no real customers; there were just some prospective customers. A development team member, who had previous experience in this domain, played the role of business expert.

Two of the rooms were used by developers who were working in two teams, Team A and Team B. These two teams were separated from each other. Beck (2000) also supports the separation of different teams. Teams A and B were close enough in adjacent rooms to communicate easily whenever required. The third room was the meeting room. In the first

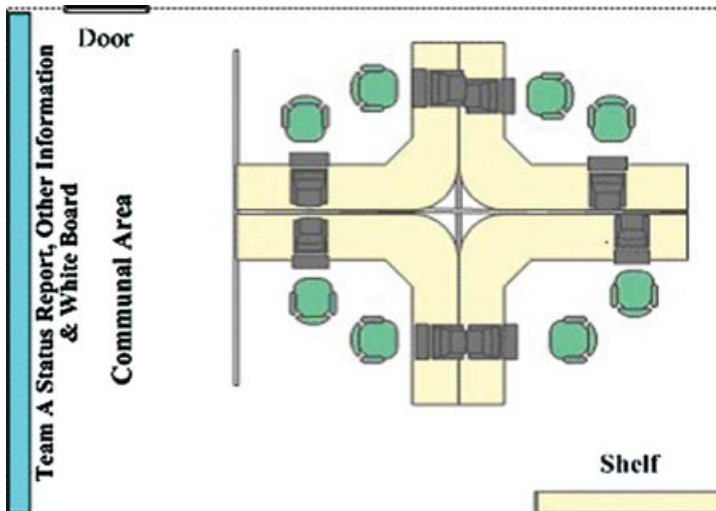


Figure 1 Room 1 layout (Mishra & Mishra, 2008).

room, Team A of the technical developers was sitting on the half-cubicle desks in groups of two people each. In the second room, business experts, the other team of developers, and other nontechnical engineers were available. These two rooms were next-door to each other so all the people working in those rooms could communicate face-to-face within a couple of seconds every time they wished. The third room was used for organizing workshops, critical releases, iterations, and other meetings. This was the room of product owners and, at the same time, the meeting room.

Other important things that was used in this project was information radiators such as project status boards, a current iteration status board, and whiteboards containing other important notes. An information radiator displays information in a place where passersby can see it. With information radiators, the passersby need not ask any question; the information simply hits them as they pass (Cockburn, 2000).

In the first room (as shown in Figure 1), near the entrance was the communal area, which was used for standup meetings by the team. At the far end past the communal area, each couple of developers was sitting on the desks that were separated by a half-height glass barrier. This arrangement increases team members' awareness of what is going on around them without using focused attention. Visual and aural accessibility are key environmental contributors to workspace awareness (Gutwin and Greenberg, 2001). According to Gutwin and Greenberg (2001), although overhearing conversations can be distracting, it may be valuable when work is highly interdependent and when the colocated people are working on the same or similar projects. Key problems of high-awareness environments include loss of privacy, loss of confidentiality, distractions, and interruptions (Brill & Weidemann, 2001; Sundstrom, Town, Brown, Forman, & McGee, 1982). However, there is some indication that a highly open environment might lead to reduced interruptions and distractions due to greater availability of nonverbal and behavioral cues that modulate interaction (Allen & Gerstberger, 1973; Backhouse & Drew, 1992; Becker & Sims, 2001). When people are focused on an individual task, their posture, eye gaze, and demeanor indicate that they are not available for conversation. However, if they look up, make eye contact, or walk around,

others are more likely to perceive them as available for interaction. Availability is largely determined by whether the person appears to be involved in focused work (Heerwagen et al., 2004). So, with this physical arrangement of workspace, Team A members could communicate with each other whenever the need arose but they also had their private space where they could concentrate on their individual tasks without interruption. Each one could see whether the other person was busy or free before interrupting. The availability of individual workspaces that aid focused attention and reduce distractions and interruption has numerous benefits (Heerwagen et al., 2004), including increased time on individual tasks (Perlow, 1999), reduced stress (Kaplan, 1992), improved performance on mental tasks (Wyon, 1996), and the ability to maintain one's line of thought and cognitive flow (Csikszentmihalyi, 1990).

Every cubicle was occupied by two members according to their roles and responsibility in the project (i.e., those couples who were working on the same part of the product or were pair programmers for some critical modules). The furniture was arranged in such a way that developer Team A could have a standup meeting or discussions whenever they wanted on a whiteboard and also could use the walls efficiently to put the diagrams, notes, module specifications, working plans, and any other documents that should be seen by other team members. It may be increasingly important to have artifacts and visual displays of information to aid coordination, memory, and understanding of the work as a whole because knowledge work activity is largely cognitive and does not leave physical traces (Lahlou, 1999; McGee, 2002).

In the communal area, one whiteboard, another board showing the status of work that the first team was doing, and one extra board for sticking other important notes needed by the first team were fixed. The whiteboard was for discussion or elaborating any story that may be erased, if required. The other board (shown in Figure 2), showing the status of the current

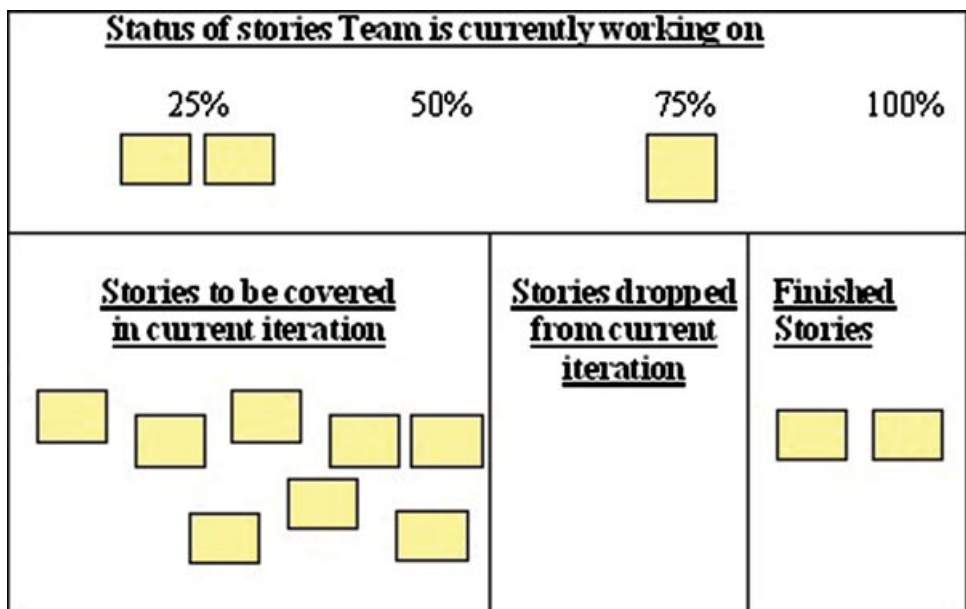


Figure 2 Status of current iteration of a team (Mishra & Mishra, 2008).



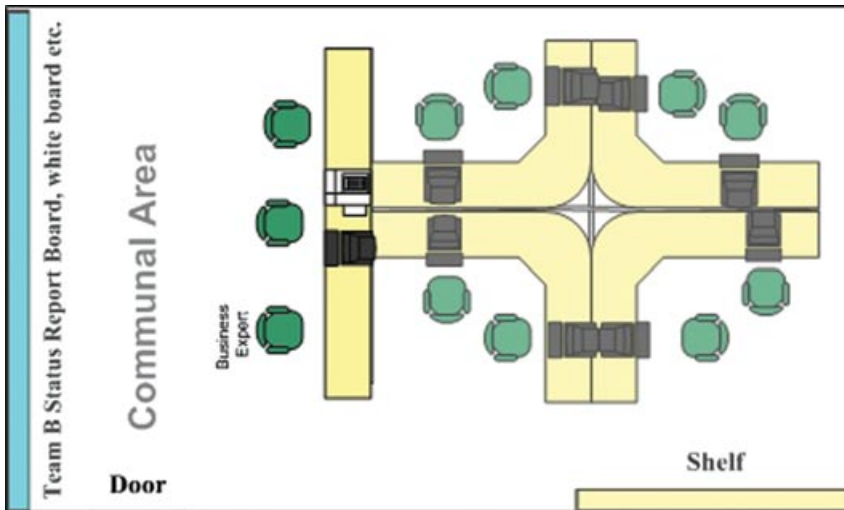


Figure 3 Room 2 layout.

iteration, had four sections. On the upper part, there was a range from 0% to 100%. Stories that this team was currently working on were stuck on this part according to percentage of work done so far on them. The bottom left corner had an area displaying stories that needed to be included in this iteration but that the team had not yet started working on. The bottom right corner contained stories that were already completed in the current iteration. The bottom middle area exhibited stories that were scheduled to be included in the current iteration but were excluded due to time constraints because some other urgent feature had come up from the user's side. The other team in the second room also had these three boards, and anyone could know the status of the work that these two teams were doing without disturbing anyone.

In the second room (as shown in Figure 3), near the entrance was the communal area. The business expert sat near the communal area at the entrance of the second room. At the far end, the second development team had the same physical seating arrangement as the first team. In this way, Team A members could talk to business expert without disturbing Team B members. In the communal area, the same three boards (one whiteboard, one board showing the status of work of the second team, and one extra board for sticking other important notes) were located on one wall. Team B could also use the communal area in this room for standup meetings using the whiteboard.

The meeting room (as shown in Figure 4) was used for the critical release, iteration, and other meetings. This was the room of the product owners and at the same time the meeting room. Here both teams' members, business expert, and other stakeholders met before each iteration or release or whenever a demo version was shown to prospective customers. This room was also used when both teams along with their business expert wanted to discuss something important all together. Additionally, this room was initially used to conduct workshops.

In this meeting room, there was a whiteboard for brainstorming and another big board showing the complete information about the status of the whole project. This board (as shown in Figure 5) was also divided into three parts. One part contained stories arranged

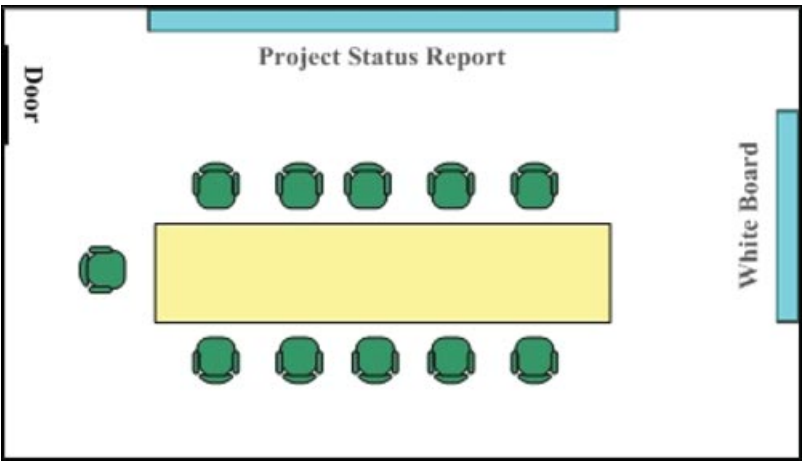


Figure 4 Room 3 (meeting room).

<u>Stories for Coming Iterations</u>	<u>Completed Stories</u>
<div>Iteration 2</div> <div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div> <div><div></div><div></div></div>
<div>Iteration 3</div> <div><div></div><div></div><div></div></div>	
<div>Iteration 4</div> <div><div></div><div></div><div></div></div>	
<div>Iteration 5</div> <div><div></div><div></div></div>	
<div><u>Unscheduled Stories or Nice Ideas</u></div> <div><div></div><div></div><div></div></div>	

Figure 5 Project status board.

in rows, each row representing subsequent iterations. The second part contained stories completed so far, along with iteration number in which they were implemented. The third section was emergent or unscheduled stories or nice ideas that are needed to be handled urgently.

4. RESEARCH METHOD AND DISCUSSIONS

We used a questionnaire to learn the effect of the new workplace environment and the use of tools like status boards and so forth on communication, collaboration, and coordination during software development. Twenty-one copies of this questionnaire were distributed mainly to software developers and customer representatives in the same small-scale software

TABLE 1. Responses to Questionnaire (in %)

Options → Question No. ↓	a	b	c	d	e
Questions for software developers:					
1	82	0	18		
2	73	0	27		
3	18	55	27		
4	55	45	0	0	0
5	64	27	9		
6	64	36	0		
7	9	91	0		
8	100	0	0		
9	18	9	73		
10	73	0	27		
11	82	9	9		
12	18	64	18		
13	73	27	0	0	
Questions for customer representatives:					
14	75	0	25		
15	0	100	0		
16	75	25	0		

development company where the work was carried out. Fifteen of twenty-one people responded, so the response rate was approximately 71%. The questionnaire is given in the Appendix, and Table 1 shows the responses for different questions. Table 2 shows the summary of responses. Later, discussions follow regarding the effect of workplace environmental factors on communication, collaboration, and coordination based on responses. Additionally, recommendations for these various factors are given.

#### 4.1. Effect of Half-Height Glass Barriers

All the team members working on one team were placed in the single room with the minimum required barrier between them to protect their ability to concentrate on their work, if required. The seating had half-height glass barriers between team members. Questions 1, 2, and 3 were asked to learn the effect of half-height glass barriers in improving communication, collaboration, and coordination and whether they increase the noise, which may adversely affect the tasks requiring individually focused attention.

Eight-two percent of software developers thought that half-height glass barriers improved the communication, collaboration, and coordination between team members, whereas 18% had no opinion on this issue. Seventy-three percent also thought that this makes them more accessible to team members and vice versa. Fifty-five percent of software developers found that, although this improved communication, it did not increase the noise that may adversely affect individually focused tasks. Eighteen percent of software developers thought otherwise. They said that these glass barriers increased the noise; therefore, they found it difficult to do individually focused tasks. Twenty-seven percent of software developers had no opinion on this issue.

TABLE 2. Summary of the Responses to Questionnaire (in %)

Options → Question No. ↓	Positive Response	Neutral Response	Negative Response
Questions for software developers:			
1	82	18	0
2	73	27	0
3	18	27	55
4	100	0	0
5	64	9	27
6	64 + 36*	0	
7	9 + 91*	0	
8	100	0	0
9	18	73	9
10	73	27	0
11	82	9	9
12	18	18	64
13	73 + 27*	0	0
Questions for customer representatives:			
14	75	25	0
15	100*	0	
16	75 + 25*	0	

\*Positive only if certain conditions exist.

So, we can say that, in general, this half-height glass barrier helps team members to do their individual work requiring concentration. At the same time, it put no limitations on communicating, coordinating, and collaborating with each other that are necessary for any creative work.

#### 4.2. Effect of Two People Sharing the Same Cubicle

Every cubicle was occupied by two members according to their roles and responsibility in the project (i.e., those couples were working on the same part of the product or they were pair programmers for some critical modules). Question 8 was asked to learn whether this arrangement helped during pair programming. As all the modules are not developed using pair programming because this is a small-scale software development organization and resources are scarce, Question 9 was asked to learn the effect of this arrangement when they are not working using pair programming or when their work is not related.

All developers agree on the fact that this seating arrangement helps during pair programming. Seventy-three percent of developers said that this seating arrangement has no effect (neither positive nor negative) on their work when they are working by not using pair programming. Eighteen percent say that it affects them positively even if their work is not related and they are not using pair programming. The reason for positive effect may be that they were motivated by each other. Nine percent said that it affects them negatively. The reason for negative effect may be that they were disturbed by each other.

So, we can say that a seating arrangement comprising two people sharing the same cubicle has tremendous positive effect during pair programming. When the work of the developers

is not related and when they are not using pair programming, then in general it has no effect on their work.

### 4.3. Effect of Status Boards

There are status boards showing information, such as status of the work, in the communal space of each team room. Anyone can get this information just by looking at the boards without disturbing team members. Questions 4, 5, 6, and 7 were asked to learn the effect of status boards in communicating the information and to find out whether these status boards contain up-to-date information and whether efforts require updating these status boards is justified. Also, these questions were asked to determine whether the boards can help in decreasing the disturbance to the team members as they all are sharing the same room to do their work.

Fifty-five percent of software developers found that the status board always contained up-to-date information whereas 45% of them said that they contained up-to-date information most of the time. Sixty-four percent of software developers thought that benefits of getting up-to-date information on the status boards outweigh the time spent on updating them whereas 27% of them thought otherwise. Nine percent of software developers had no opinion on this issue.

Sixty-four percent of software developers always preferred getting the information about the status of other developers' work from status boards whereas 36% preferred to get the information from status boards when the information is not related to their work. Nine percent of developers always preferred to ask other developers about the status of their work rather than getting it from the status board. Ninety-one percent preferred to ask other developers only when their work was closely related to other developers and information was not yet available on the board.

So, we can say that status boards play an important role in nonverbal communication. They display the information that does not require further analysis, and this reduces the disturbance that may be caused otherwise.

### 4.4. Effect of Communal Space in Each Room

There was a small space for standup meetings in each room. This space also contained whiteboards so that developers could clarify their ideas by writing and drawing diagrams, which are sometimes needed in addition to verbal communication. Question 10 was asked to learn the effect of this space on overall team communication, collaboration, and coordination. Question 11 was posed to learn whether developers prefer standup meeting in this space or whether they prefer to have a proper meeting even if the issue that needs to be discussed is related to their team only.

Seventy-three percent of developers said that having a communal space in their team's room enhanced communication, collaboration, and coordination among team members. Eighty-two percent of developers preferred a standup meeting in this space if the issue was related to their team only.

So, we can deduce that communal space is a useful space that can improve communication, collaboration, and coordination among team members. In some offices, there are other communal spaces like coffee machine space. If an organization has such kind of space, then the communal space given in our case study can be reduced but cannot be replaced. Communal space given in our case study has multiple functions. It enhances the informal

face-to-face communication between software developers. It also helps the developers to clarify some complex issues that cannot be solved by only verbal face-to-face communication by using tools like whiteboards. Face-to-face communication at a whiteboard is a “hotter” communication medium than is face-to-face communication (Ambler, 2002). Also, this space is used for standup meetings, which may require status boards, and so forth. Other communal space (like coffee machine space) can support only informal communication between developers.

#### **4.5. Effect of Separating Different Teams in Different Rooms**

Every team, working on a different part of the same software, was put in a separate room. Separating two teams from each other, but still allowing them to be at a close distance from each other, helped in reducing disturbance that might have been caused if they had been placed in the same room. If these teams were put in the same room and if one team wanted to have a standup meeting to discuss some issues related to its part, it would have caused disturbance for other team members. Teams must communicate and collaborate with each other whenever necessary, however. This was carried out in the third room, which was also used as a meeting room for all stakeholders (management, developers, customers, people from sales, etc.). Questions 12 and 13 were asked to find out whether separation of teams in different rooms deteriorated communication, collaboration, and coordination among teams or whether it helped in enhancing the focus of each team on its part of the project.

Sixty-four percent of developers thought that separation of teams did not effect communication, collaboration, and coordination among them. The reason may be that the projects that have been done so far were modular in nature and these projects had well-defined parts that were not interrelated closely. Also, rooms of these teams were close enough that the distance did not effect communication, collaboration, and coordination among them. They also used a third formal meeting room for meeting between these teams. In the case of a project with low modularity and highly related parts, separation of teams may pose challenges in communication, collaboration, and coordination especially if the rooms for these teams are not adjacent (e.g., on different floors).

#### **4.6. Effect of This New Arrangement on Customer Representatives**

Questions 14, 15, and 16 were asked to customer representatives to find the effect of this new arrangement on them. Seventy-five percent of customer representatives said that developers were more accessible for them due to new half-height glass barriers. They did not benefit from status boards much, as all of them said that they preferred using a status board only when they did not have any issues to clarify. Seventy-five percent of them preferred talking directly to the team leader, irrespective of whether they had some issues to clarify.

This clarifies that although the new workplace arrangements make developers more accessible to customer representatives, keeping the up-to-date information on different status boards is not beneficial for them. They prefer to have direct verbal communication even if they just need some static information.

### **5. CONCLUSION**

Communication, collaboration, and coordination are significant components of software development in general and specifically to agile methodology of software development,

which is becoming popular. It has already been proven that communication, collaboration, and coordination not only facilitate the software development process, but also ensure quality and productivity. In this article, we have presented the comprehensive literature survey along with its observation while developing complex software using agile methods. Based on our study, we strongly believe that physical environment and the effective use of tools like whiteboards, statusboards, and so forth, played an important role in communication and collaboration. Separate cubicle areas helped in performing tasks that require considerable focus and attention, whereas half-height glass barriers between these cubicles made their occupants visible and reachable to each other and also facilitated in communication, coordination, and collaboration. A communal area displaying the status of the team's work and other important details facilitated productive discussion among team members when required. Close proximity of business experts and separate but adjacent rooms for two different teams working on the same project also helped in inter- and intra-team communication, collaboration, and coordination. In summary, our findings provide important insights that are valuable for small organizations using agile methods. For practitioners, the recommendations of this study offer that productivity and better quality can be accomplished by adopting the right workplace environment in such a way that can facilitate excellent communication, collaboration, and coordination among all stakeholders.

This study will help as a guideline for small and medium software development organizations toward improving quality, productivity, and timely delivery of software to their customers. For researchers, this article points out new directions on how to proceed with the study of various other approaches of agile methods. Further empirical studies and comparison in terms of quality, productivity, and delivery time in different settings/environments will be areas of further research in this direction.

## 6. APPENDIX

Questionnaire for assessing the effect of physical environment and tools like whiteboards, status boards etc. on communication, collaboration and coordination between team members during software development

PART A (for development team members)

1. Do you think that half-height glass barriers improve communication, collaboration, and coordination between team members?  
a) Yes      b) No      c) Can't say
2. Do you think that half-height glass barriers make you more accessible to other team members and vice versa?  
a) Yes      b) No      c) Can't say
3. Do you think that half-height glass barriers increase the noise and make it difficult to do individual focused tasks?  
a) Yes      b) No      c) Can't say
4. Do the status boards display up-to-date information?  
a) Yes      b) Most of the time      c) No      d) Rarely      e) Can't say

5. Do you think that the benefits of getting up-to-date information on the status board outweigh the time spent updating the status board?
  - a) Yes
  - b) No
  - c) Can't say
6. Do you prefer getting the information about the status of other staff members' work from the status board?
  - a) I always prefer getting the information from status board.
  - b) I prefer getting the information from the status board only when my work is not related to the work of other members.
  - c) Can't say
7. Do you prefer asking other staff about the status of their work?
  - a) I always prefer asking staff members.
  - b) I prefer asking only when my work is closely related to their work and information is not yet available on the status board.
  - c) Can't say
8. Do you think that present seating arrangement comprising two people sitting in one cabin helped during pair programming?
  - a) Yes
  - b) No
  - c) Can't say
9. In what way did the present seating arrangement, comprising two people sitting in one cabin, affect your work when you were not working using pair programming?
  - a) Positively
  - b) Negatively
  - c) No effect
10. Do you think that communal/discussion space for standup meetings helps in communication, collaboration, and coordination?
  - a) Yes
  - b) No
  - c) Can't say
11. Do you prefer standup meetings in communal/discussion space in your room or do you prefer using a third room (formal meeting room) to clarify the issues related to your team?
  - a) I prefer standup meetings in communal space if the issues are related to my team only.
  - b) I prefer formal meetings in the third room even if the issues are related to my team only.
  - c) Can't say
12. Do you think that separation of different teams in different rooms deteriorates the communication, collaboration, and coordination between different teams?
  - a) Yes
  - b) No
  - c) Can't say
13. Do you think that separation of different teams in different rooms helps each team to focus on its part of the project and avoids unnecessary confusion?
  - a) Yes, always
  - b) Yes, but only when the project can be properly modularized
  - c) No
  - d) Can't say



## PART B (for customer representatives)

14. Do you think that the new seating arrangement having half-height glass barriers makes development team members more accessible to you?
  - a) Yes      b) No      c) Can't say
15. Do you prefer getting the information about the project from status boards?
  - a) I always prefer status boards.
  - b) I prefer status boards only when I don't have some issues to clarify.
  - c) Can't say
16. Do you prefer getting information directly from team leaders?
  - a) I always prefer talking directly with team leaders.
  - b) I prefer talking directly with team leaders only when I have some issues to clarify.
  - c) Can't say

**ACKNOWLEDGMENT**

We thank the Editor-in-Chief and reviewers for their valuable comments and directions to improve the quality of this article.

**REFERENCES**

- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). Agile software development methods: Review and analysis (p. 107). Espoo, Finland: VTT Publications. Available at: [www.inf.vtt.fi/pdf/publications/2002/P478.pdf](http://www.inf.vtt.fi/pdf/publications/2002/P478.pdf)
- Allen, T. J. (1971). Communications networks in R&D laboratories. *R&D Management*, 1(1), 14–21.
- Allen, T. J., & Gerstberger, P. J. (1973). A field experiment to improve communications in a product engineering department: The non-territorial office. *Human Factors*, 15(5), 487–498.
- Ambler, S. (2005). Quality in an agile world. *Software Quality Professional*, 7(4), 34–40.
- Ambler, S. W. (2002). Validating agile models. *Cutter IT Journal*, 15, 33–39.
- Backhouse, A., & Drew, P. (1992). The design implications of social interaction in a workplace setting. *Environment and Planning B: Planning and Design*, 19, 573–584.
- Beck, K. (2000). *Extreme programming explained: Embrace change*. Upper Saddle River, NJ: Addison-Wesley.
- Becker, F., & Sims, W. (2001). *Offices that work: Balancing communication, flexibility, and cost*. International Workplace Studies Program. Ithaca, NY: Cornell University. Available at: <http://iwsp.human.cornell.edu>
- Boehm, B. W. (1987). Improving software productivity. *Computer*, 20(9), 43–57.
- Boger, E. (2003). Electronic tracking board reduces ED patient length of stay at Indiana Hospital. *Journal of Emergency Nursing*, 29, 39–43.
- Bostrom, R. B., & Thomas, B. D. (1983). Achieving excellence in communications: A key to developing complete, accurate and shared information requirements. *Proceedings of the Twentieth Annual Computer Personnel on Research Conference* (pp. 1–13) November 17–18, 1983, Charlottesville, VA.
- Brill, M., Weidemann, S., & BOSTI Associates (2001). *Disproving widespread myths about workplace design*. Jasper, IN: Kimball International.
- Cao, L., & Balasubramaniam, R. (2007). Agile software development: Ad hoc practices or sound principles? *IT Professional*, 9(2), 41–47.
- Cockburn, A. (2000). Agile software development. Draft version 3b available at <http://unjobs.org/authors/alistair-cockburn>
- Cockburn, A. (2002). *Agile software development*. Indianapolis, IN: Addison-Wesley.
- Csikszentmihalyi, M. (1990). *Flow: The psychology of optimal experience*. New York: Harper & Row.

- Curtis, B., Curtis, H., & Iscoe, N. (1988). A field study of the software design process for large scale systems. *Communications of the ACM*, 31(11), 1268–1287.
- Daft, R. L., & Lengel, R. J. (1986). Organizational information requirements, media richness and structural design. *Management Science*, 32, 554–571.
- Daft, R. L., Lengel, R., & Trevino, L. K. (1987). Message equivocality, media selection, and manager performance: Implications for information support systems. *MIS Quarterly*, 11, 355–366.
- Duncan, T., & Moriarty, S. E. (1998). A communication-based marketing model for managing relationships. *Journal of Marketing*, 62(2), 1–13.
- Edstrom, A. (1997). User influence and the success of MIS projects: A contingency approach. *Human Relations*, 30, 580–607.
- Elliott, J. J. (2000). Design of a product-focused customer-oriented process. *Information and Software Technology*, 42(14), 973–981.
- Finsterwalder, M. (2001). Does XP need a professional customer. XP2001 workshop on customer involvement, in XP2001, May 21, 2001, Sardinia, Italy. Available at <http://homepages.cwi.nl/~arie/wci2001/finsterwalder.pdf>
- France, D. J., Levin, S., Hemphill, R., Chen, K., Rickard, D., Makowski, R., et al. (2005). Emergency physicians' behaviors and workload in the presence of an electronic whiteboard. *International Journal of Medical Informatics*, 74(10), 827–837.
- Greening, J. (2001). Using XP in a big process company: A report from the field. Paper presented at XP Universe Conference, July 23–25, 2001, Raleigh, NC.
- Gutwin, C., & Greenberg, S. (2001). A descriptive framework of workspace awareness for real time groupware. CSCW'01. Dordrecht, The Netherlands: Kluwer.
- Heerwagen, J. H., Kampschroer, K., Powell, K. M., & Loftness, V. (2004). Collaborative knowledge work environments. *Building Research & Information*, 32(6), 510–528.
- Highsmith, J. (2002). *Agile software development ecosystems*. Indianapolis, IN: Addison Wesley.
- Jain, A. K., & Ting, P. D. (1989). Software quality via rapid prototyping. *Global Telecommunications Conference, 1989, and Exhibition. Communications technology for the 1990s and beyond. GLOBECOM '89*. Institute of Electrical and Electronics Engineers, 1, 642–646.
- Jensen, J. (2004). United hospital increases capacity usage, efficiency with patient-flow management system. *Journal of Healthcare Information Management*, 18, 26–31.
- Kaplan, R. (1992). Urban forestry and the workplace. In P. H. Gobster (Ed.), *Managing urban and high-use recreation settings*. General technical report NC-163. North Central Forest Experiment Station, USDA Forest Service, Chicago.
- Kendon, A. (1990). *Conducting interaction: Patterns of behavior in focused encounters*. Cambridge, UK: Cambridge University Press.
- Korkala, M., Abrahamsson, P., & Kyllonen, P. (2006). A case study on the impact of customer communication on defects in agile software development (pp. 76–88). *Agile Conference*, July 23–28, 2006, Minneapolis, MN.
- Kraut, R. E., & Streeter, L. A. (1995). Coordination in software development. *Communications of the ACM*, 38, 69–81.
- Lahlou, S. (1999). Observing cognitive work in offices. *Proceedings of the Second International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*. N. A. Streitz, J. Siegel, V. Hartkopf, & S. Konomi (Eds.), October 1–2, 1999, Lecture Notes In Computer Science, 1670. Springer-Verlag, 150–163.
- Larman, C. (2003). *Agile & iterative software development*. Pittsburgh: Addison-Wesley Professional.
- Lei, D., Slocum, J. W., & Pitts, R. A. (1999). Designing organizations for competitive advantage: The power of unlearning and learning. *Organizational Dynamics*, 27, 24–38.
- Marinakakis, H. A., & Zwemer, F. L. Jr. (2003). An inexpensive modification of the laboratory computer display changes emergency physicians' work habits and perceptions. *Annals of Emergency Medicine*, 41, 186–190.
- McGee, J. (2002). Knowledge work as craft work. McGee's musings. Available at: [www.mcgeesmusings.net/stories/2002/03/21/KnowledgeWorkAsCraft.html](http://www.mcgeesmusings.net/stories/2002/03/21/KnowledgeWorkAsCraft.html)
- McLuhan, M. (1964). *Understanding media: The extension of man*. New York: McGraw Hill.
- Melnik, G., & Maurer, F. (2004). Direct verbal communication as a catalyst of agile knowledge sharing. *Proceedings of the Agile Development Conference (Adc'04)*, Volume 00 (June 22–26, 2004). ADC. Washington, DC: IEEE Computer Society, pp. 21–31.

- Mishra, D., & Mishra, A. (2008). Workspace environment for collaboration in small software organization. In *Proceedings of the 5th International Conference on Cooperative Design, Visualization, and Engineering*, September 21–25, 2008, Calvia, Mallorca, Spain, LNCS, 5220, 196–203.
- Pashler, H., Johnson, J. C., & Ruthruff, E. (2001). Attention and performance. *Annual Review of Psychology*, 52, 629–651.
- Perlow, L. A. (1999). The time famine: Toward sociology of work time. *Administrative Science Quarterly*, 44, 57–81.
- Ribbens, B. A. (1997). Organizational learning styles: Categorizing strategic predispositions from learning. *International Journal of Organizational Analysis*, 5, 59–73 (January).
- Rico, D. F. (2008). Effects of agile methods on website quality for electronic commerce (pp. 463–471). *Proceedings of the 41st Hawaii International Conference on System Sciences*, January 7–10, 2008, Waikoloa, Big Island, Hawaii.
- Rising, L., & Janoff, N. S. (2000). The Scrum software development process for small teams. *IEEE Software*, 17, 26–32.
- Schein, E. H. (1993). How can organizations learn faster? The challenge of entering the green room. *Sloan Management Review*, 34, 85–95 (Winter).
- Sharp, H. C., & Robinson, H. M. (2003). An ethnography of XP practice. *Proceedings of the co-located 15th Annual Psychology of Programming Interest Group Workshop and Empirical Assessment of Software Engineering Conference* (pp. 15–27), April 8–10, 2003, Keele University, Staffordshire, UK.
- Sinnott, M. J. (1990). An aid to the management of pediatric trauma in peripheral centers—a proposal for a pediatric trauma board. *Journal of Emergency Medicine*, 8, 413–417.
- Spencer, R., Coiera, E., & Logan, P. (2004). Variation in communication loads on clinical staff in the emergency department. *Annals of Emergency Medicine*, 44, 268–273.
- Sundstrom, E., Town, J. P., Brown, D. W., Forman, A., & McGee, C. (1982). Physical enclosure, type of job, and privacy in the office. *Environment and Behavior*, 14(5), 543–559.
- Vosburgh, J., Curtis, B., Wolverton, R., Albert, B., Malec, H., Hoben, S., et al. (1984). Productivity factors and programming environments. *Proceedings of the Seventh International Conference on Software Engineering*, March 26–29, 1984, Orlando, Florida.
- Walston, C. E., & Felix, C. P. (1977). A method of programming measurement and estimation. *IBM System Journal*, 16.
- Wears, R., Perry, S., Wilson, S., Galliers, J., & Fone, J. (2007). Emergency department status boards: User-evolved artefacts for inter- and intra-group coordination. *Cognition Technology & Work*, 9(3), 163–170.
- Wu, L., & Sahraoui, H. (2005). Accommodating software development collaboration. *Proceedings of the 12th Asia-Pacific Software Engineering Conference (Apsec'05)*, Volume 00 (December 15–17, 2005). APSEC. Washington, DC: IEEE Computer Society, pp. 33–42.
- Wyon, D. P. (1996). Indoor environmental effects on productivity. *Proceedings of IAQ'96, "Paths to Better Building Environments"*, October 6–8, Baltimore, MD.
- Xiao, Y., Schenkel, S., Faraj, S., Mackenzie, C. F., & Moss, J. (2007). What whiteboards in a trauma center operating suite can teach us about emergency department communication. *Annals of Emergency Medicine*, 50(4), 387–395.