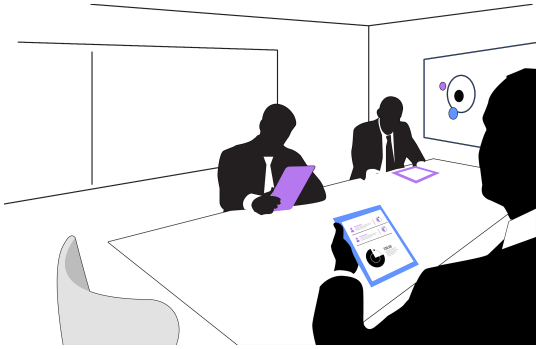
TÉCNICO LISBOA



VI-TRUST: Visualization and Interaction on project "Transitions to the Urban Water Services of Tomorrow"

João Miguel Pataca dos Santos Delgado Oliveira

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Joaquim Armando Pires Jorge
Prof. Alfredo Manuel dos Santos Ferreira Júnior

Examination Committee

Chairperson:	Prof. João Emílio Segurado Pavão Martins
Supervisor:	Prof. Joaquim Armando Pires Jorge
Member of the Committee:	Prof. Daniel Jorge Viegas Gonçalves

November 2014

To my friends, for making me enjoy the ride.

To my family, for building the ride.

And to Mafalda, for keeping the ride in its right track.

Acknowledgments

A word of gratitude to all the professors who accompanied me throughout the years at Instituto Superior Técnico. A lot of them were great role models, some others made me want to pull my hair off, but all of them made me learn something. A special thanks to Prof. David Matos for making me sure that this course was the right decision, to Prof. Paulo Carreira for making me dream big, and to Prof. Alberto Sardinha for giving me the opportunity to work at INESC.

Another special thanks is due to my supervisor, Prof. Joaquim Jorge, for giving the direction on what eventually became this thesis, and to my co-supervisor, Prof. Alfredo Ferreira, for patiently revising the text with me.

I would like to thank YDreams, in particular Antão Almada and Mónica Pedro, for inviting me to work with them. I must also thank all my friends there, namely João Batista, Vasco Verdes and Xavier Faure, who were crucial in helping me develop the systems used in this dissertation.

All of this wouldn't be possible without my friends, who were there during good and bad times: Alexandre Quitério, Ana Waddington, Daniel Magarreiro, David Afonso, Edgar Santos, Emídio Silva, Joaquim Guerra, Miguel Vaqueiro, Pedro Pinto, Ricardo Toureiro, Rodrigo Bruno, Sara Boiça and many others.

I would also like to thank my family, for supporting me all this time, even when I decided to do some crazy stunts (going to Japan and all that). In particular, I would like to thank my father for making me feel I was making the right decisions, my mother for being there no matter what, my sister Catarina for showing me the value of pursuing my dreams and my sister Mafalda, her husband Pedro and my nephews Francisco and João for guidance and for bringing joy to everyone's lives.

Though all of these people played an important role in bringing me to this point, there is one in particular whose influence was bigger than anything else. I thank you, Mafalda Valente, for fascinating me every day and for making me want to impress you.

Resumo

Colaborar através de software vem sendo explorado há bastante tempo, mas continuam a existir barreiras na interação que a tecnologia não permite ultrapassar. Em algumas situações acontece mesmo que as pessoas rejeitam software para reuniões presenciais e preferem usar ferramentas sem software.

Colocamos como hipótese que uma das razões para isto acontecer é que sistemas que utilizam apenas meios tradicionais de interação não permitem detetar meios naturais de comunicação entre pessoas, dos quais destacamos a utilização do espaço. O estudo deste tipo de comunicação é conhecido como proxemics e propomos utilizá-lo para melhorar software de reuniões presenciais. No nosso sistema, os utilizadores apresentam as suas contribuições simplesmente mudando de posição na sala. Este movimento é algo que se faz mesmo sem software - para apresentar, as pessoas movem-se para a frente da sala, onde podem captar o foco dos restantes participantes. Quando alguém se coloca nessa posição, todos entendem o que significa, mas os computadores precisam de ser informados através de inputs explícitos por parte dos utilizadores.

Com um sistema capaz de detetar relações espaciais isto não acontece - o sistema adapta-se ao mesmo tempo que os utilizadores mudam de posição, não havendo necessidade de mais input. Para avaliar o impacto da utilização de proxemics, construímos dois sistemas com a mesma funcionalidade e adicionámos proxemics a um deles. Ao testar ambos com utilizadores verificámos que proxemics permite uma redução de interações explícitas, o que por sua vez é responsável por maiores índices de satisfação por parte dos utilizadores.

Palavras-chave: proxemics, CSCW, interfaces naturais, HCI

Abstract

Collaborative software has been explored for a long time, but still there are obstacles to interaction that technology has been unable to overcome. In some situations, people discard face-to-face meeting software and go back to using non-software tools.

We hypothesize that one of the reasons for this to happen is that systems using only traditional means of interaction cannot capture natural means of communication between people, among which are spatial relationships. The study of this type of communication is known as proxemics and we propose using them for improving face-to-face meeting software. In our system we allow users to present and highlight their work simply by moving in the room. This movement is something that people do anyway, even when not using software – to present work, users move to the front of the room, where they can easily capture the focus of everyone. When someone moves in that position, all of the people in the room understand what that means, but computers must be informed with explicit input from a user.

With a system that is aware of the spatial relationships, this doesn't happen – the system adapts as users change position, requiring no other input. To evaluate the impact of proxemics, we built two systems with the exact same functionality and add proxemics to one of them. We tested both systems with users and our results show that proxemics leads to less explicit interactions, which in turn yields better user satisfaction.

Keywords: proxemics, CSCW, natural interfaces, HCI

Contents

List of Tables

List of Figures

Glossary

CSCW Computer Supported Cooperative Work is the use of technology to support people in their work.

HCI Human-Computer Interaction is a field of study in computer science targeted at the design and implementation of the interfaces between humans and computers.

IAM Infrastructure Asset Management is the set of knowledge and strategies for managing public infrastructure assets like water treatment facilities, sewers, bridges, etc.

NUI Natural User Interface refers to an interface that takes advantage of natural interactions (body movement, gestures, etc.) as opposed to artificial interactions (e.g. mouse and keyboard).

Proxemics A set of theories related to the use of space in non-verbal communication. It focuses mainly on the dynamics of personal and social space, space territories and cultural differences.

TRUST Transitions to the Urban Water Services of Tomorrow is an European project targeted at achieving a sustainable, low-carbon water future without compromising service quality.

Ubicomp Ubiquitous computing is a concept that associates computing power and intelligence to virtually any object. In this sense computers can be everywhere, therefore ubiquitous.

Chapter 1

Introduction

In this chapter we begin by detailing the motivation behind our work in section 1.1. We then explain how that motivation leads to the main challenge for this dissertation and what is our main research goal in section 1.2. Through the achievement of that goal we can provide two important contributions, which are discussed in section 1.3. Finally, we present the remaining document structure in section 1.4.

1.1 Motivation

Software for supporting human collaboration has been in use for a long time, but in face-to-face meetings, people are still feeling limited by computers. The truth is that humans communicate through means that are not typically perceived by machines, in particular spatial relations. Even though we don't realize it consciously, we interact with each other through very simple spatial cues that everyone understands. For instance, something as simple as walking in someone's direction indicates that we wish to speak with that person.

Because the software we use is not capable of perceiving this implicit communication, it forces users to inform the system through explicit interactions, like the press of a button. This results in people feeling like the computer is in their way, instead of supporting their tasks as it was intended, because they are forced to repeat something they have already communicated.

As such, there is an added protocol present in the software that wasn't there before. Because of this, users tend to ignore the system, using older methods and not effectively adopting the solutions developed for them.

1.2 Objectives

The challenge for this dissertation is to address the issue identified in the previous section for a particular scenario - a face-to-face meeting targeted at raising new ideas and making decisions for the TRUST European project - Transitions to the Urban Water Services of Tomorrow[1].

[1] Project Reference: 265122; Website: http://www.trust-i.net/

Early in this document we identify the recurring problems of applications with similar purposes - commonly referenced as CSCW (computer supported cooperative work). We speculate that these problems can be solved through the adequate application of proxemics in the design and implementation of CSCW software. Proxemics is the study of the use of space in human communication and recently it has been increasingly adopted in human-computer interaction.

Our main research goal is to understand whether proxemics can effectively reduce the number of explicit interactions (mouse clicks, taps, gestures) with the computer. We hypothesize that through proxemics it is possible to communicate more information with less explicit interactions. With the ability to understand the use of space implemented, users are required to spend less effort interacting and there is a reduction in the protocol for collaboration.

1.3 Contributions

The main contribution of this work is understanding the impact that proxemics can have in CSCW. We have developed a solution for a particular scenario but our work can be reproduced for similar situations. Through the comparison of two versions of our system whose only difference was that one took advantage of proxemics, we have observed the differences in how the software is used and how that affects user satisfaction. We have then discussed those differences and how they address the recurring issues of CSCW applications.

In our solution we equipped the system with the ability to detect when users wish to present their contributions. Taking advantage of the fact that people present their work in the front of the room, we detect when users approach that area and adapt the whole system in accordance.

Because the scope is limited to a face-to-face meeting we have been able to accurately identify where using proxemics makes sense. We have experimented with several alternatives for the same function and have evaluated user feedback on each of them. This discussion on why certain alternatives work or not provides an important basis for other people designing CSCW applications aware of proxemics information.

1.4 Document Structure

In the remaining document we present how we understood our challenge, how we approached it and how we evaluated our contributions. First we discuss the related work in chapter 2, where we explore the recurring problems of CSCW and how using proxemics might provide a solution to those. In chapter 3 we explain how we approached those problems in our scenario and how other works on proxemics have influenced our design. The development process was integrated in an internship at YDreams[2], which was the company responsible for producing the contribution to the TRUST project. That process was based on user-centered development guidelines explained in chapter 4, which allowed us to have everything in place for understanding the influence of proxemics in our scenario. That influence was

[2] http://www.ydreams.com

evaluated in chapter 5, which explains how we compared two versions of our software - one with and other without proxemics. Also in that chapter we discuss the differences we observed and why that happened. Lastly, we conclude in chapter 6, presenting a final discussion on our work and on what could be done in the future.

Chapter 2

Related Work

Building a natural interaction for a collaborative software requires knowledge in both CSCW and Human Interaction - be it with other humans or with machines. The intersection of these areas yields the scope of this dissertation, which will be explained in the following sections.

In the CSCW section the scope of the project will be analyzed in the context of CSCW. By analyzing previous work in the same scope it is possible to understand what problems remain unsolved.

The Proxemics section is a review of work on proxemics in isolation - understanding how humans use space to communicate with each other.

The following section, Proxemics applied to HCI, consists of an exploration of work by other researchers that, knowingly or not, have used proxemics in their studies. In this section, a set of works is highlighted as an inspiration for what can be done with proxemics.

Lastly, we present a brief discussion on all of these topics.

2.1 CSCW

In order to define the scope of the work to be produced, it is important to identify where it fits in the CSCW matrix (Figure 2.1). Due to a preliminary analysis of the users' needs with our partners it was decided to focus only on the "same place, same time" quadrant. Support for a situation where two teams would be collaborating at the same time in different locations was considered but it was dropped in order to keep focus on the most important quadrant.

Table 2.1 sums what can be expected from the relationship between people in this environment - the key characteristics of collocated synchronous interactions. The main conclusion of the analysis of this table is that users in this context already have an established means of communication much more effective than anything that can be done with current technology. As such, the application should be seen as a way to mediate and support the communication, without ever getting in its way.

Table 2.1 becomes even more relevant when analyzed together with Grudin's work on why CSCW applications fail (Grudin [13]). Grudin identifies three problems that arise from a lack of understanding of how people perform their work: the disparity between who does the work and who gets the bene-

Characteristic	Description	Implications
Rapid feedback	As interactions flow, feedback is as rapid as it can be	Quick corrections possible when there are noticed misunderstandings or disagreements
Multiple channels	Information among participants flows in many channels – voice, facial expressions, gesture, body posture, etc.	There are many ways to convey a subtle or complex message; also provides redundancy
Personal information	The identity of contributors to conversation is usually known	The characteristics of the source can be taken into account
Nuanced information	The kind of information that flows is often analog or continuous, with many subtle dimensions (e.g., gestures)	Very small differences in meaning can be conveyed; information can easily be modulated
Shared local context	Participants have a similar situation (time of day, local events)	A shared frame on the activities; allows for easy socializing as well as mutual understanding about what's on each others' minds
Informal "hall" time before and after	Impromptu interactions take place among subsets of participants upon arrival & departure	Opportunistic information exchanges take place, and important social bonding occurs
Co-reference	Ease of establishing joint reference to objects	Gaze and gesture can easily identify the referent of deictic terms
Individual control	Each participant can freely choose what to attend to, and change the focus of attention easily	Rich, flexible monitoring of how all of the participants are reacting to whatever is going on
Implicit cues	A variety of cues as to what is going on are available in the periphery	Natural operations of human attention provide access to important contextual information
Spatiality of reference	People and work objects are located in space	Both people and ideas can be referred to spatially; "air boards"

Table 2.1: Key characteristics of collocated synchronous interactions as identified by Olson and Olson [23].

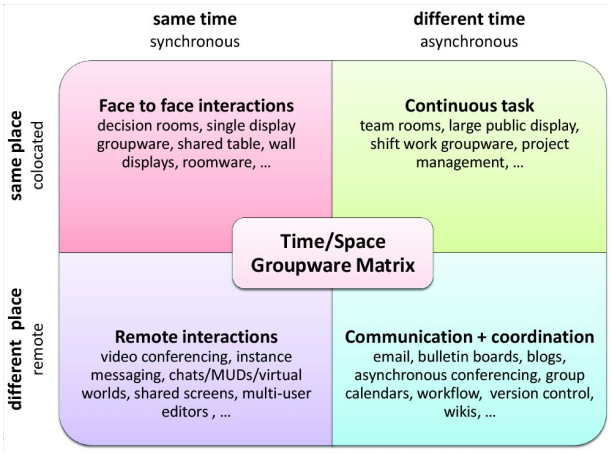


Figure 2.1: The CSCW matrix, defined by Johansen [18].

fit; the breakdown of intuitive decision making; and the underestimated difficulty of evaluating CSCW applications.

Dourish and Bellotti [9] further identified three problems with traditional collaborative software in regards to awareness - defined by the authors as ”an understanding of the activities of others, which provides a context for your own activity”. The first problem is that the user who provides the information is not the one who benefits from it (note the consistency with the analysis from Grudin [13]). Secondly, individuals may benefit from others’ contribution but this is not guaranteed. What happens is that individuals receive the contributions that the initiator considers appropriate. And thirdly, delivery of information is controlled more by the sender than by the recipient.

The authors then proceed to propose a new approach based on shared feedback. The main difference is that the information on others’ activities is displayed in a shared workspace, not in each of the private ones. This approach overcomes the problems identified earlier and takes full advantage of having more than one screen available.

2.2 Proxemics

Proxemics was introduced in the sixties by Edward T. Hall as the ”study of man’s perception and use of space” (Hall [14]). In this text, Hall focused on the different ways one perceives and uses space based on one’s culture. Perhaps one of his most fundamental conclusions is that there is no pattern that can be applied to every individual - different people don’t react in the same manner to the same stimuli.

Consider a situation where two friends are talking face to face - the distance between them will vary depending on their culture. This concept is illustrated in Figure 2.2. It happens because people with the same culture share ”an elaborate and secret code that is written nowhere, known by none, and understood by all” (Sapir [29]).

However, it is important to stress that the fact that two individuals share the same culture does not mean they will have the same reaction. In fact, according to a study carried out by Au and Cheung [3], ”the effect of the ICV (intra-cultural variation) is independent of and similar in magnitude to that of the

Figure 2.2: Differences in personal space in two different cultures.

cultural mean". What can be concluded is that it is more likely for two individuals to react in similar ways if they share the same culture, but differences must always be expected.

This is supported by work from Gerhart and Fang [11]. They re-analyzed data gathered by Hofstede's study on international differences in work-related values (Hofstede [15]) and discovered that the respondents' company accounted for more variance than the respondent's country.

Hall [14] later developed the field of proxemics when he identified categories for the distance between two people (see Figure 2.3). According to Hall, the social relationship between two individuals is directly related to physical distance, which means that it is possible to estimate the interactions that might emerge from the distance category at which people position themselves.

Public Distance (3.7m and up) is the distance formed by people when hearing a public speaker. At this distance, verbal and nonverbal communication must be amplified in order to convey the same meaning as subtleties in smaller distances. Voice volume is higher, enunciation more precise and gestures are more pronounced.

Social Distance (1.2 to 3.7m) is the distance reserved for small social interactions. If someone stands at this distance, we are compelled to interact, which means that going in and out of this distance is a common means of seeking someone's attention. For example, when addressing someone we don't know for asking directions we stand at a social distance, but before and after, we stand at a higher distance, signaling no intent for disturbance.

Personal Distance (46 to 120cm) is a distance at which typically only close acquaintances, friends and relatives stand. At this point, physical contact is possible and, as such, it can be used by someone trying to exert dominion or control over someone else.

Intimate Distance (touching to 46cm) is usually reserved only for lovers. The presence of someone at such close distance is perceived by unmistakable sensory inputs: highly detailed vision of the other person, the smell and the body heat.

Another important contribution to proxemics are F-Formations (Kendon [21]). According to the au-

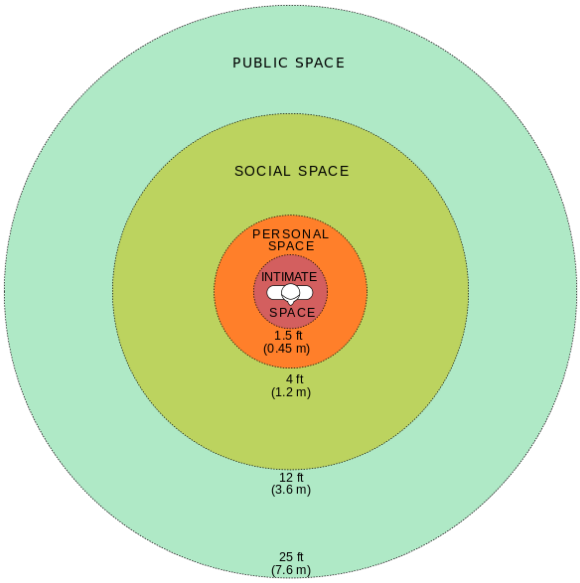


Figure 2.3: Different personal spaces distinguished by distance, according to Hall [14].

thor, people tend to organize themselves in spatial formations when interacting, creating a shared space where they can communicate. This has two main purposes: to allow the participation of every individual involved and to separate the group from other individuals not taking part in the interaction. The formations are based on how people organize three distinct spaces: O-space - the observed space - every individual stands around this shared space, facing it; P-space - the participants' space - the area surrounding the O-space where the participants are located; R-space - the outer space, which is the area that surrounds the group but is still under the influence of the participants. People that are not part of the group tend not to get too close (crossing the R-space).

Based in the position and orientation of the group's participants in these spaces, Kendon identified the following formations: Vis-à-vis - participants are face to face (restricted to two participants); L-shape - the angle between participants is approximately 90°; Side-by-side - the angle between participants is approximately 0°; and Circle - participants are organized in a circle.

In our scenario, F-Formations are particularly important because we are interested in detecting when people want to share information. This happens when they are facing the audience, which is organized around a meeting table, defining an approximate Circle formation - everyone is involved in the discussion and there is a person in particular who is leading it.

2.3 Proxemics applied to HCI

One of the most influential works in this area is the one by Nicolai Marquardt and Saul Greenberg, applying the concepts of proxemics to human-computer interaction and ubiquitous computing. Instead of having devices that are oblivious to their surroundings, they were able to create a system that provides the devices with info on which users and other devices are around them. Nicolai's thesis provided three main contributions: a proxemics framework, the Proximity Toolkit and three case studies.

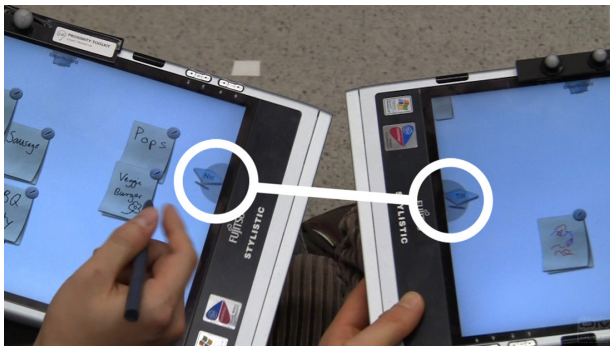


Figure 2.4: Example of interaction in Till Ballendat's proxemic brainstorming software - sharing content between devices.

The proxemics framework is targeted at designers building interaction around proxemics. The author defined five key dimensions to consider when designing - distance, orientation, movement, identity and location. He also identified a common strategy when building this kind of interactions - the gradual engaging pattern -, which defines how a system should react to user proxmity, moving between awareness, reveal and interaction. Both of these contributions are a direct product of what has been achieved in the proxemics field and that has been summarized in the previous section.

The Proximity Toolkit provides a service for detecting proxemic relationships between devices and/or users. With this toolkit it becomes very simple to develop proxemic interactions - there is only the need to subscribe to events provided by the service and react to them accordingly.

The three case studies are explorations of the design of systems using proxemics. Because of the novelty of this approach, these studies provide valuable information.

One of these case studies is a collaborative brainstorming tool, developed by Till Ballendat (Figure 2.4). In spite of being only a proof of concept, it already shows how multiple devices can be used in benefit of such sessions. It allows users to create virtual sticky notes on their private screens and share them in a public screen. Being a demo, this system provides a great number of ideas that can be further explored.

Table 2.2 is a summary of possible proxemic interactions for information visualization, as identified by Jakobsen et al. [17]. In this article, three user studies are conducted in order to understand which proxemic interactions work for visualizing information. The most promising scenarios, studies 1 and 2, involved a simple adaptation of the visualization to the user distance and orientation. In user study 1, for navigating a map, the users would step forward and backward to zoom in and out. As for user study 2, users could aggregate information on a map based on three distance limits - which is in accordance with the gradual engaging pattern (Greenberg et al. [12]). The first one is out of the scope of this thesis - it is something that can already be done with more precision using multi-touch devices. As for the second one, we used it but adapted it for changing visualization in a shared screen. The same principle is applied, but considering a different task - instead of visualization and navigation, this thesis is more focused on information organization (rightmost column of Table 2.2).

Another example that takes advantage of proxemics is Range, developed by Ju et al. [19]. Range provides an interactive white-board that allows different interactions depending on the distance of the

	Visualize	**Filter**	**Sort**	**Select**	**Navigate**	**Coordinate**	**Organize**
Distance	Show details when close / aggregates when far	Filter items depending on the physical distance to user (body fisheye)	-	Distance increases selection scope	Focus and demagnified context at distance	Brush-and-link close data	Distance dependent workspaces
Orientation	Visualize for different viewing angles	-	Sort by variable selected by orientation	Coarse selection by orientation	Head orientation controls zoom center	Indicate related areas through orientation	-
Movement	Switch between encodings by moving	Dynamic querying when moving	Sort by variable selected by movement	Coarse selection by movement	Zoom and pan by moving relative to display plane	Selected views move along with user	Reorganize windows in workspace
Location	Contextual visualiza-tions	Switch between subsets	-	-	Overview and detail in left to right	-	Location dependent perspec-tives or activities

Table 2.2: Combination of information visualization tasks and proxemics categories as identified by Jakobsen et al. [17].

users. The distances and corresponding interactions were derived from work by Hall [14] and are again in accordance with the gradual engaging pattern identified by Greenberg et al. [12]. Furthermore, this paper provides observations gathered during the iterative development that can be used as a starting point. For instance, the authors observed that if multiple users were standing near the white-board at different distances, the interaction mode should be derived from the distance of the user that is closer. The main difference between Range and the work in this thesis is that instead of using a screen as a white-board, we use it as a support for presentation and communication of the current stage of the session.

Also on the subject of white-board based applications, work by Rekimoto [26] on using hand-held computers together with a shared screen to improve the interaction is extremely relevant. Rekimoto first identifies the design problems of white-board only approaches like inputing text, getting personal data and multitasking. He then shows how these problems can be addressed with the help of a hand-held computer, producing several solutions. Considering this work was produced in 1998, before the rise of the smartphone and the tablet, taking advantage of current technology should allow for even more solutions using multitouch screens together with mobile devices. Rekimoto [25] also documented a fundamental gesture that seems to be reasonable in a proxemics-based system - the pick-and-drop - which involves selecting an object in one screen and dropping it in another by tapping an empty area. This gesture is also in the same lines as that described by Seewoonauth et al. [33] except this one relies on Near Field Communication (NFC) tags. Both methods can be further improved with current technology and with proxemics. Instead of using a pen, a finger, or the device itself to ”pick” and then to ”drop”, the users could make a grabbing gesture and then a release gesture in the direction where they intend to drop the content. We considered this kind of gestures when integrating proxemics in our solution but realized that there was no situation where it would make sense in our scenario.

For understanding how Rekimoto’s and others’ work came to fruition it is relevant to look at applications built for the Xerox Liveboard (Elrod et al. [10]), in particular the Tivoli (Pedersen et al. [24]), which is one of the first white-board like applications based on a large interactive screen. The work on the Tivoli provides not only a valuable source of interface ideas but also the reasoning behind them. It’s this reasoning that remains relevant to this day, for even though technology might have evolved, the tasks the users want to perform should remain similar, if not the same.

Other works with slightly different goals may also serve as an inspiration for the interaction with a shared screen. Morris et al. [22] provide a look into how interactive tabletops might be a tool for collaborative search tasks. Also on tabletops, Dohse et al. [8] have successfully merged the multitouch data from a tabletop with hand tracking data to enhance the multiuser interaction. It’s particularly interesting how they used it not only to expand the possible interactions but also to improve the touch detection. Depending on the hardware available this is a technique that might be required still today.

In the tabletops field, the Medusa authors (Annett et al. [2]) describe how the hardware works but what is really interesting is the explored interactions. For instance, they experiment with assigning different functions to different sides of the table and on how to inform the users that the system recognizes their presence. It is not clear if all the interactions described actually produce good results, but Medusa

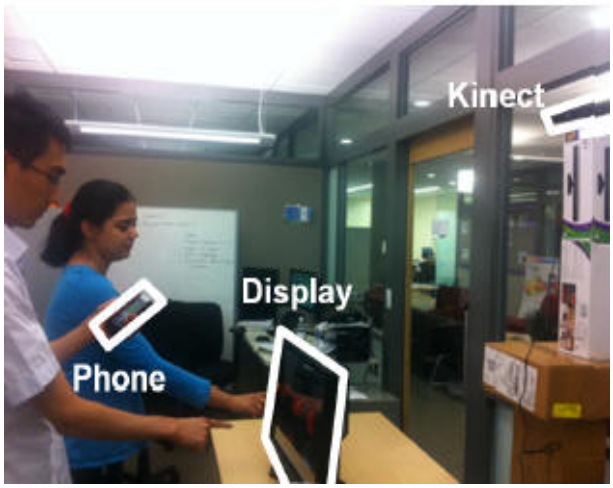


Figure 2.5: Using accelerometer data to associate a user with a touch (Rofouei et al. [27]).

is definitely an inspiration for what can be produced in this thesis.

Work by Scott et al. [32] on territoriality in collaborative tabletop workspaces also provides interesting ideas. The author identified three territories - personal, group and storage -, that define which interactions should be supported by the table. The work also includes design guidelines for building tabletop interactions which can be relevant for any proxemics project using this device.

For multiuser tabletop interactions it might be needed to identify the user that is producing each touch. The work on DiamondTouch from Dietz and Leigh [7] is quite ingenious - they have antennas transmitting an electric signal on the table and a receptor connected to each user. The signal indicates the location of the touch while the receiver indicates the producer of the touch. At the moment it is possible to achieve a similar result but with a different method - using depth sensors. This would not only be a simpler solution, it would also allow supporting more dynamic settings with an undetermined number of users and with dynamic locations.

There is also a great amount of work on identifying the user who is interacting with a screen. Rofouei et al. [27] correlates the mobile device's accelerometer data and depth data from a depth sensor to associate a user to a touch on the screen (Figure 2.5). A similar method is described by Hutama et al. [16] but without a depth camera, using instead the phone itself to touch the surface. Schmidt et al. [30] also use the mobile device's accelerometer for turning the phone into a pointer and to further expand the possible interactions between the screen and the mobile device. As for Schöning et al. [31], they propose a somewhat complex method for authenticating users, involving detecting touches both in the screen and in the phone and then using the mobile's camera flash together with a Bluetooth communication, allowing identification. These techniques are all interesting solutions for scenarios in this thesis that require user identification when interacting with the screen.

Not only identifying the user but also pairing devices is increasingly relevant. This is something that has been researched for interactions with public screens, as shown in an analysis by Rukzio et al. [28] of the usage of mobile phones for personalized interactions with ubiquitous public displays. The author's distinguish different categories of personalization and interaction and arrive at a solution that involves

a web server which is public and which knows the location of the screen. The user then needs to take a picture of a marker on the screen and send it to the server. This way, the user is able to personalize what appears on the screen. These steps need to be repeated every time because it is never known with which screen the user will be interacting with. However, in a meeting room environment, this will remain constant. Therefore this method could be used but adapted for one use - the first moment the user arrives at the room.

For understanding how to support different user formations in the room, the work by Cristani et al. [5] is an important reference. The authors were able to detect F-formations using unobtrusive computer vision techniques and concluded that the results of their system were consistent with proxemics findings: social and physical distance are directly related. This work is particularly interesting for learning how to detect F-formations, which are required for adapting the interaction to moments when groups of users are together.

Designing the interactions can never be completely separated from the design of the place where these interactions take place. In building a meeting room system, special care must be taken in how the room is arranged. The book by Sommer [35], ”Personal Space: The Behavioral Basis of Design”, provides a means to take advantage of spatial arrangement, affecting the user's behavior towards interaction. In our case we took advantage of a long table and a space near the screen to encourage users to present their work.

The very same principle is present in a work by Cooperstock et al. [4] where the authors have iteratively designed a meeting room. The paper describes how the room was arranged to support different activities and also how the iterative development was handled.

2.4 Summary

From the analysis of CSCW work, it is clear that this software fits in the ”same place, same time” quadrant of the CSCW matrix (Johansen [18]). This placement implies a variety of characteristics from Table 2.1 by Olson and Olson [23], from which we realize the importance of allowing users to communicate freely without forcing them to use the software. Analyzing this table and understanding how people work in this situation allows us to know how to address the reasons why CSCW applications fail (Grudin [13]): the disparity between who does the work and who gets the benefit; the breakdown of intuitive decision making; and the underestimated difficulty of evaluating CSCW applications.

Being a ”same place, same time” system means that it must also deal with problems related to awareness, as identified by Dourish and Bellotti [9]. We follow a suggestion made by these authors, basing our solution in a shared screen and personal devices. This way, users can receive information from others if they want to, but they are also free to work on their personal matters without a big distraction.

Understanding proxemics makes it possible to address some of these concerns, namely the breakdown of intuitive decision making and providing awareness. In general terms, it should allow for a more 'natural' interaction. For understanding exactly what 'natural' means, this chapter included a section for exploring the principles of proxemics, focusing on the theories of how people interact using space.

In our scenario we take advantage of the moment when people approach the shared screen for presenting their work. Therefore we are interested in changing visualization depending on where people are relative to the screen. Hall's theory on personal spaces provides clues to how we should approach this problem. More than using the distances suggested by Hall, what is important in this scenario is to identify at what distance to the shared screen a participant becomes a speaker. This can't be labeled easily - it depends on cultural backgrounds and mostly on the physical disposition of the room.

In this case, the developed system is for a meeting room that will be used by people of the same company. This means that even though they might not be part of the exact same culture, they still share a background: the same work environment, the same location, the same socio-economical environment, etc. We encountered a rather homogeneous group of users but we always took special care to notice differences in background.

This system will be used in different companies, however, which means it needs to be easily adapted, not just because of different cultures, but also because of different spaces. This was addressed by making all the trigger distances adjustable, allowing different companies to use different settings according to their need. According to research on F-Formations, approaching the screen is not enough - we should also consider whether the person is facing the audience or not.

Applying proxemics to HCI is relatively new and has recently received a great improvement with Marquardt's contribution. Marquardt created a framework for designing interactions using proxemics and a toolkit that allows easy deployment of proxemic software. To validate his work, he also provided three examples of the use of this toolkit. This and other works on white-board and tabletop applications indicate that interactive systems benefit from designing with proxemic interactions in mind. In most of the works mentioned in the previous section we immediately recognize potential, even without thinking too much on why that happens. This is because we can tell that the way people interact with machines in those works is much closer to the way we interact with people.

What is not so clear is the benefit of this type of interactions in real applications - in our case a face-to-face meeting. Which interactions make sense in this scenario? How much of the interaction should be done through more traditional methods or through proxemics? And how can we measure the impact of proxemics on the way users perceive the system?

In the following chapter we explain our approach to provide an answer to these questions.

Chapter 3

Using Proxemics to Improve Collaboration

From the analysis of related work it is clear that CSCW applications have a set of recurring problems and proxemics might provide solutions. In the previous chapter we identified four main problems with CSCW applications (based on work by Grudin [13] and Dourish and Bellotti [9]): 1. the disparity between who does the work and who gets the benefit; 2. the breakdown of intuitive decision making; 3. the underestimated difficulty of evaluating CSCW applications; and 4. dealing with awareness.

We consider that two out of these four problems can be addressed by developing a system around proxemics: the breakdown of intuitive decision making and providing awareness.

The breakdown of intuitive decision making happens because CSCW software adds protocol to tasks that can be carried out in simpler terms using traditional resources. By having a system that detects the spatial relations between people and devices, it is possible to take advantage of how people interact with each other - by changing the state of the system at the right moment we are able to simplify the protocol.

Improving awareness is also something that can and has been done in other proxemics solutions. It is precisely one of the main advantages of using the gradual engaging pattern, as identified by Marquardt and as put into practice in the Range white-board. Because the system reacts to the user's use of space, there is an opportunity for a better control of what the user interacts with. This directly improves awareness of the system in regards to people, which in turn allows people to be more aware of what is happening in the meeting - the system is always showing information correctly adapted to the situation, therefore people can easily be informed just by checking the shared and/or their own screen.

In this chapter we discuss our approach to developing a face-to-face meeting software that targets the CSCW problems through the use of proxemics. In section 3.1 we discuss the context of this project - what scenario we must support and what work has been done previously. We then move on to briefly explain how our solution supports this scenario without proxemics in section 3.2. We also discuss the software architecture of our solution in section 3.3, which leads to how we equipped the system with proxemics, presented in section 3.4. Lastly, we explain how we tackled skeleton detection for taking advantage of proxemics in section 3.5.

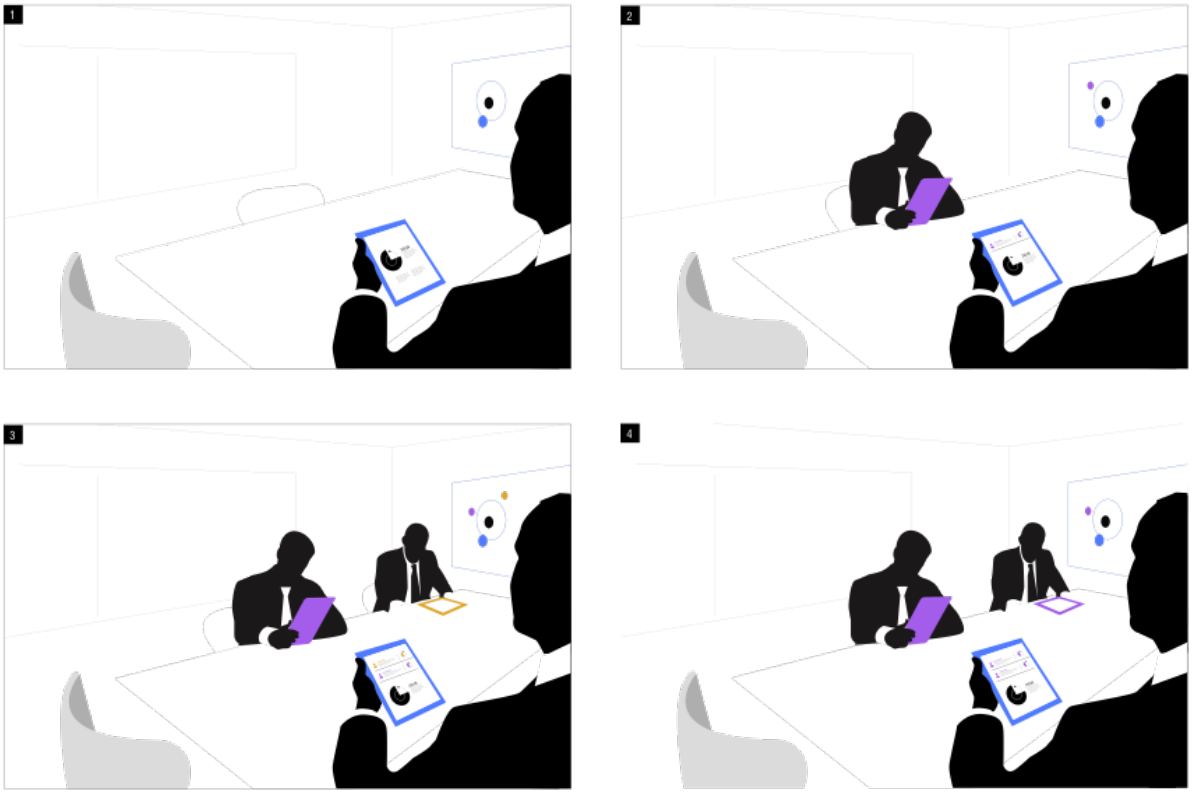


Figure 3.1: Sharing information using only each users' device.

3.1 Context

Before the beginning of the development process, YDreams' proposal was to create an immersive environment designed for collaboration among multiple decision-makers / stakeholders. It would address the technological barriers often created by specialized tools in industry decision-makers, at the various levels. Improved human engagement would be leveraged by a variety of tools to improve decision-making and collect participant input and collaboration. This environment would act as a highly expressive and interactive front-end for information and models developed in TRUST, bridging the gap between the information produced across the TRUST portfolio and the common manager and decision-maker.

It also had two clearly defined goals: to design a decision theater to support collaboration among decision-makers and stakeholders, allowing multiple users, together and simultaneously, to explore and share information; and to explore new paradigms of natural user interfaces in order to avoid the traditional human-computer interaction limitations in these types of scenarios.

Figure 3.1 and Figure 3.2 show the first abstract ideas for the software, focusing mainly on how information was shared between people and devices.

The meeting process (summarized in Figure 3.3) was also provided in the beginning of the development by the partners of the project, who have been undertaking this sessions using white boards, pen and paper for a few years. The steps are heavily based on the work by Alegre and Coelho [1] on Infrastructure Asset Management (IAM) of Urban Water Systems:

1. Writing down the mission and vision, which should be the starting point for the whole decision

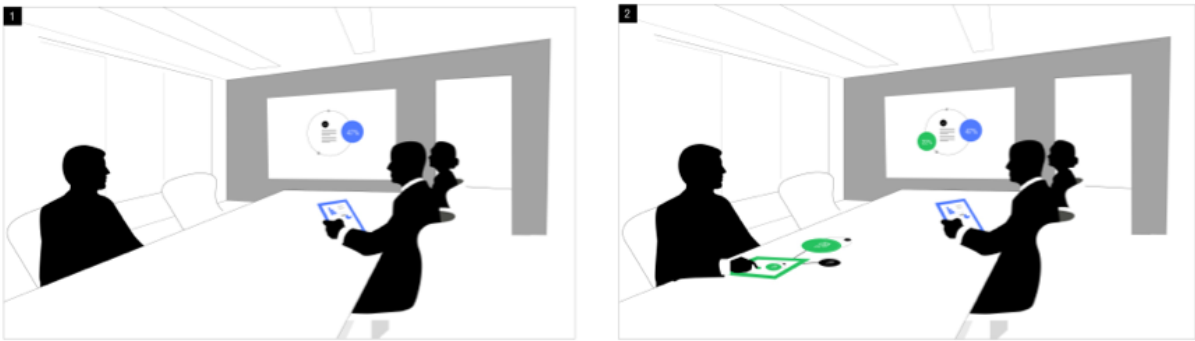


Figure 3.2: Sharing information using a multi-touch table and a screen.

Figure 3.3: The process which the software must support.

process and should be decided prior to the meeting. Every proposal and every action during the meeting should take the mission and vision into account;

2. Participants propose objectives and work on their text until a consensus is reached.

   (a) Each objective must be assigned to a TRUST dimension: Economic, Social, Environmental, Assets or Governance. If one objective does not clearly belong to a dimension, it should be rewritten.

   (b) Each objective should be compared and validated with the objectives suggested by TRUST in 'A Master Framework for UWCS Sustainability'. If any of the objectives is similar, participants may choose the TRUST version, as it provides a standard text and a set of criteria.

3. For each objective, participants propose criteria. At this point, they might conclude that the objectives need to be redefined. If so, the participants should return to step 2.

4. For each criterion, participants propose metrics. When doing this, users may use metrics contained in a metrics library (as available in the AWARE-P software). Again, they might realize that the previous definition of objectives and criteria is not adequate. In that case, they should go back to step 2 or 3 accordingly.

5. Every metric should be assigned a weight (0.5, 0.75, 1, 1.5 or 2). This weight represents a relative measure – for instance, if metric A weights 2 and metric B weights 1, this means that metric A is twice as important as metric B. The weight of a criterion is the sum of the weight of its metrics. In the same way, the weight of an objective is the sum of the weight of its criteria. Participants strive to reach a balanced result, in which no objective is much more important than the others. If this happens, there might be a need to redefine metrics, criteria and/or objectives.

6. Once the participants are satisfied, they proceed with the planning process in AWARE-P, using the results of the meeting. In the decision theater software this is done by generating an Excel file that can be imported in AWARE-P.

3.2 Base System

Our goal is to understand whether we can use proxemics to improve CSCW software, particularly in our scenario. But that does not mean that all of the interactions must be based on proxemics. In fact, that is a problem of some of the work previously done in this area - it might be interesting to explore from an interaction point of view, but it is clear from a CSCW perspective that it would not be practical in this context. For example, using the mobile phone as a pointer (as in work by Schmidt et al. [30]) introduces some interesting interaction possibilities, but it is not something that makes sense for this context, where we can support the users' tasks better with more traditional alternatives.

Basing our final solution mostly in proven interaction methods assures that we produce a valid system - one that could be used by real companies. This in turn raises the value of our results, as it shows

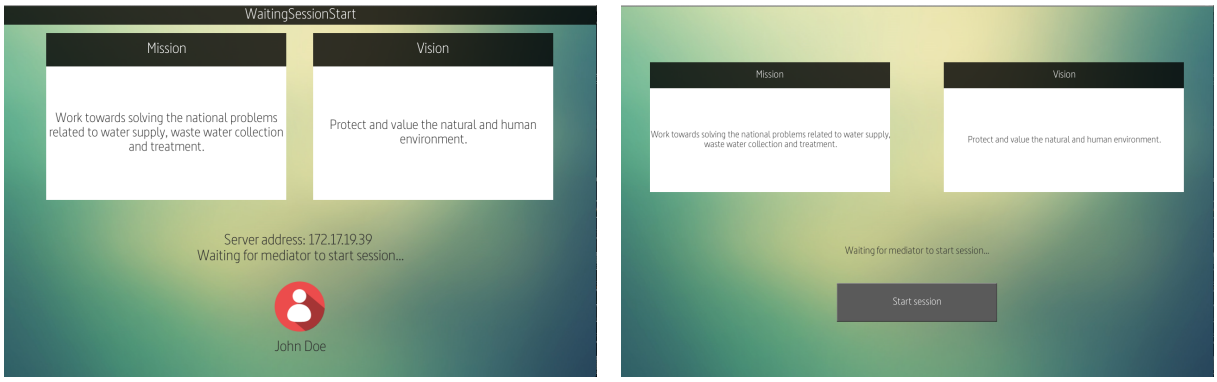


Figure 3.4: Screenshot of mission and vision insertion screen. On the left, the shared screen and on the right the mediator screen.

that using proxemics in the right situations introduces improvements in real applications, not just in hypothetical prototypes.

Our solution consists of a distributed system based on a shared screen and personal devices (tablets and/or personal computers). This screen always shows the current focus of the meeting and it is controlled by a mediator through his own device (either a tablet or a personal computer). Other users can freely navigate the system and can submit their contributions at different moments during the meeting. Every person can see everyone else's contribution, but only the author and the mediator can change it. Every contribution is distributed through all the devices and the shared screen. This solution of having personal and shared spaces is based on work by Dourish and Bellotti [9] and Rekimoto [26].

It is important to note that we considered one other alternative for our system using a tabletop instead of a shared screen (based on work by Annett et al. [2] and others in this field). However, we realized that this solution was not targeting our users' needs properly and abandoned it for a shared screen alternative.

The shared screen process must be the first to start. All users are then able to join the session by introducing their names. As soon as the mediator joins, he is able to edit the mission and vision (Figure 3.4) which, as mentioned in the previous section, should be defined before the meeting.

Figure 3.5: Screenshot of the table of objectives, criteria and metrics.

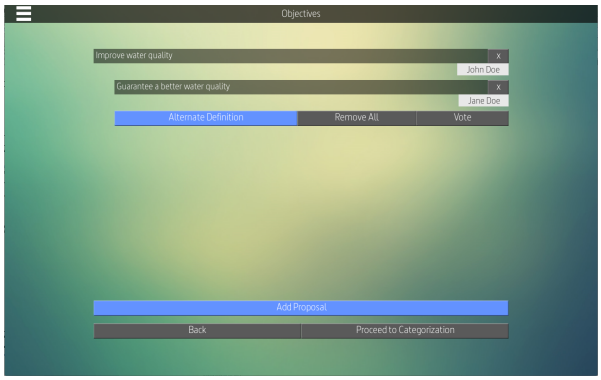


Figure 3.6: Screenshot of proposal mode.

The main goal of the session is to fill the table of objectives, criteria and metrics (Figure 3.5). This table is filled by entering proposal mode, where users can submit their proposals and alternatives (Figure 3.6). The users are allowed to vote on their favorite proposals, to import proposals from files and to validate their work by comparing their proposals to the ones suggested by TRUST.

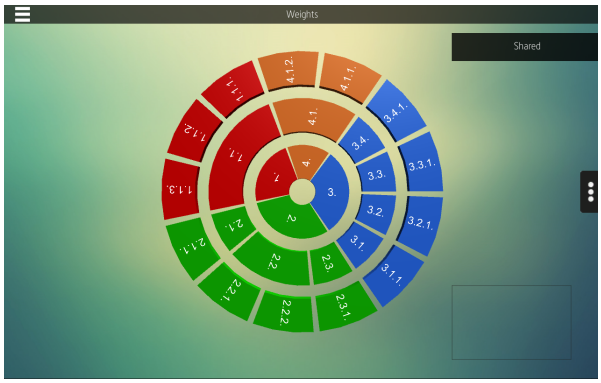


Figure 3.7: Screenshot of weight assignment.

Once all users agree on the goals, criteria and metrics, the next step is to assign a relative weight to each metric. To visualize the impact of the assignment, we have introduced a dynamic circular chart, as seen in Figure 3.7. Each user is able to make a personal weight distribution and it is up to the mediator to decide which one is better.

In order to save the work and to later use it in the following steps of the planning process, it is possible to export the results of the meeting to an Excel file. This file can later be imported by our system and it was designed to be used by other software in TRUST project.

Before explaining the architecture, it should be mentioned that we took care in developing an aesthetic language that would be in accordance with what is mostly in use at the moment (newest versions of Windows, iOS and Android, for instance) with the goal of making users comfortable with using the system. For this reason the main inspiration were flat design guidelines - minimal design elements, minimal use of 3D elements (the only exception were the weight distribution charts) and a color palette composed mainly of highly saturated and bright colors.

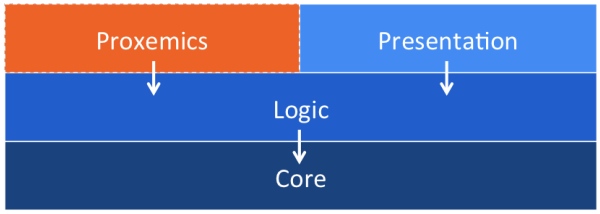


Figure 3.8: The software architecture.

3.3 Software Architecture

All of these features are supported by the architecture illustrated in Figure 3.8. It is a layered architecture comprised of three main layers: Core, Logic and Presentation. A fourth optional layer is added at the same level as the presentation layer for dealing with proxemics.

The Core contains the main elements manipulated by the system. This includes the items (objectives, criteria and metrics) and the proposals and alternatives, which are a simplified version of items.

The Logic layer is the one that controls the state and transitions in the application. It manipulates the elements of the Core according to the current state and it exposes controls for changing it. The Logic module also handles network synchronization - all devices are aware of a global state and each of them also have a local state. This way users can easily check information on their devices without losing track of what is the current focus of the discussion.

The Presentation layer presents content according to the current state of the Logic layer and it is able to use the controls exposed by this layer to request state changes.

Finally, the optional Proxemics layer interacts only with the Logic layer by interacting with only a subset of controls - the ones that control visual adaptation according to proxemics information. It is important to note that these controls are also available to the Presentation layer, allowing access to the same functionality even without proxemics. This is what we use to be able to compare the two versions of the system.

3.4 Equipping the System with Proxemics

As previously discussed, we used proxemics only in a very limited range of situations, valuing quality over quantity. We carefully identified the moments of the meeting that could benefit from a system aware of spatial relations and focused only on those. See section 4.6 for a discussion on how we identified these moments and how we approached them.

The first moment where we introduced proxemics was in the discussion of proposals and alternatives. In this moment, users are presenting their ideas and it is normal to see differences in opinion. From our tests with users we realized this was a good candidate for proxemics - people who want to defend their position tend to move nearer to the shared screen, where they can easily get everyone else's focus.

For the same reason, the other place where we used proxemics was in the weight assignment stage. This stage is slightly different from the previous one, as it takes a while for users to work on their contribution, but the end goal is the same - to present and discuss a contribution to the remaining users.

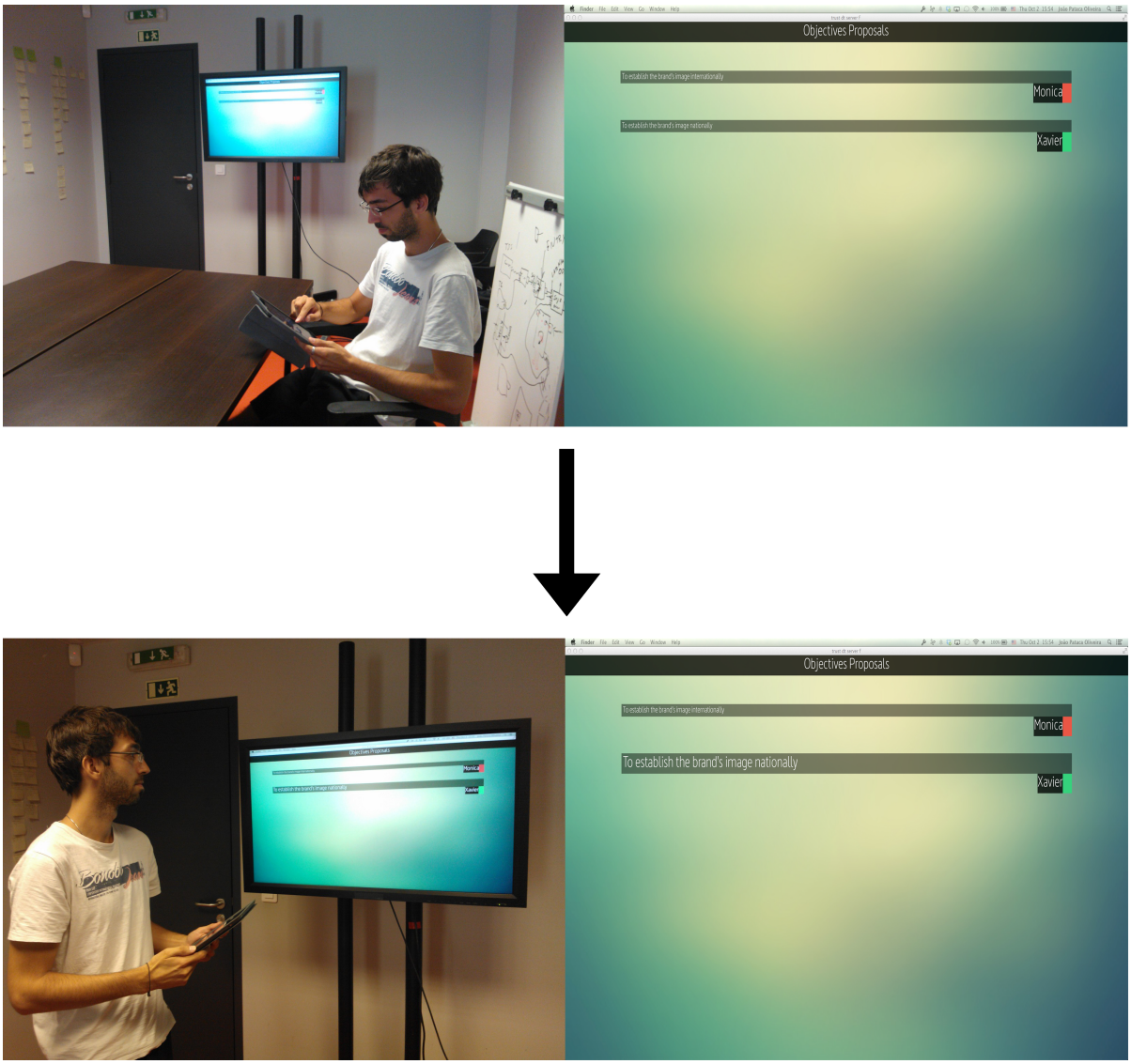


Figure 3.9: Visual adaptation in proposals screen.

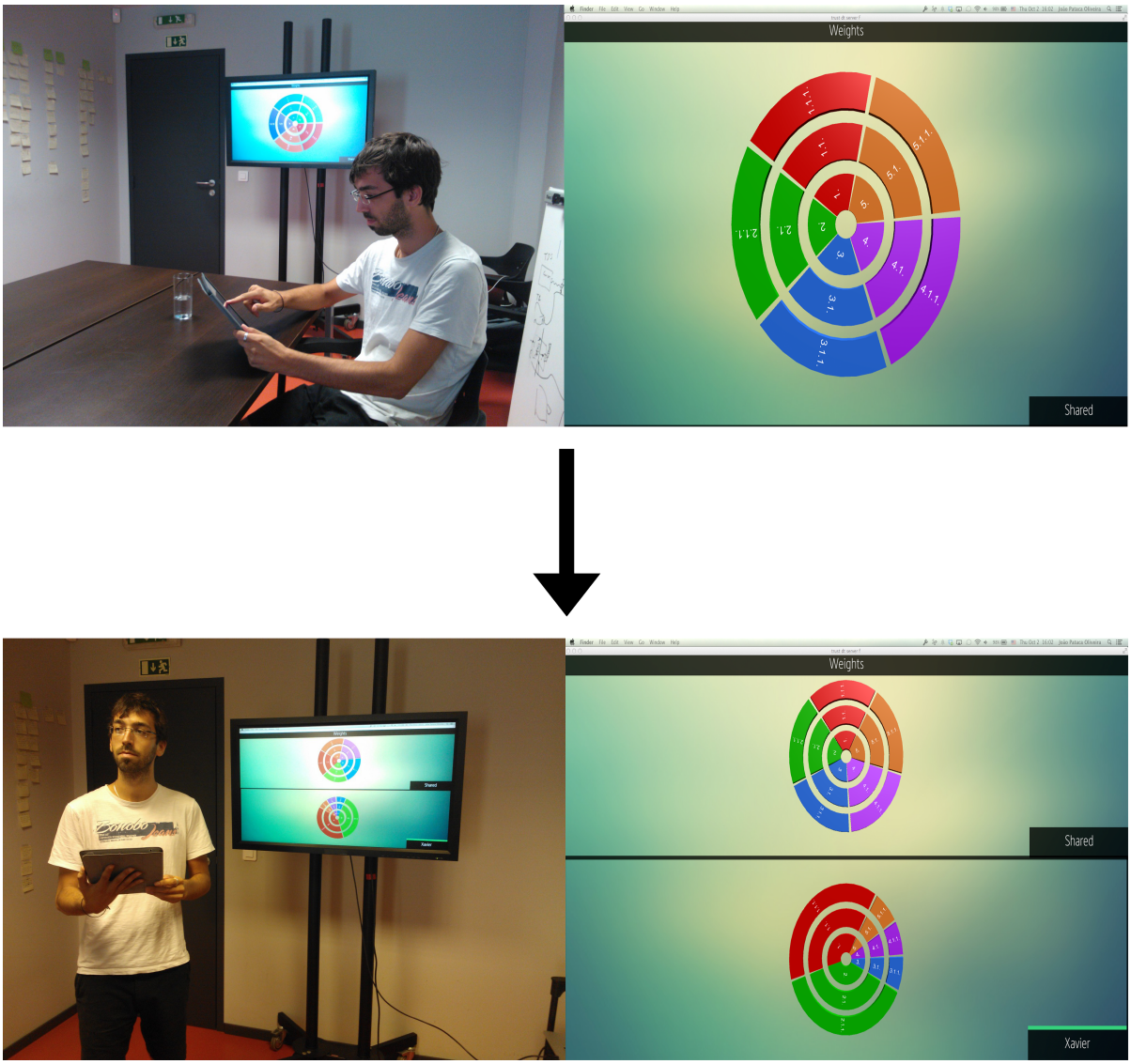


Figure 3.10: Visual adaptation in weights screen.

To take advantage of the natural tendency to approach the shared screen, we detect when this happens. If the system is in proposal mode, it adjusts the size of the proposal of a user - the closer a user is to the screen, the bigger his contributions will appear. This behavior is illustrated in Figure 3.9. Note that this change is visible in the shared screen and in every user's device.

As for weight discussion, we have implemented a similar system - when users are at a distance lower than a certain value (depending on the room setup) their weight distribution is shown on everyone's screen (Figure 3.10). To prevent situations where users are standing at the limit distance and the system is constantly triggering a change in visualization, the distance for removing a contribution is slightly higher than the distance for adding a contribution.

Both of these mechanisms are directly targeted at the CSCW problems we previously identified: awareness and the breakdown of intuitive decision making.

By constantly highlighting what is currently in discussion without any explicit input from the users, everyone in the room can easily recognize what is going on, just by looking at the shared screen or their

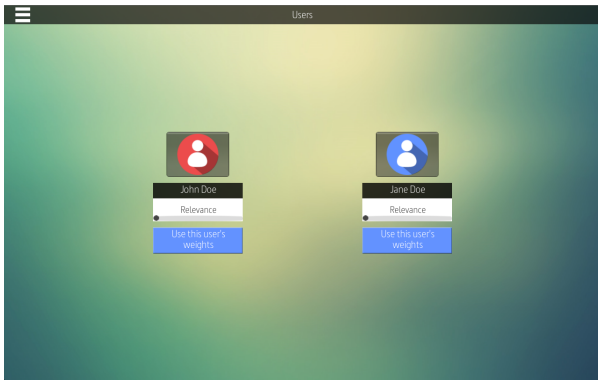


Figure 3.11: Controlling the contributions highlight in the non-proxemics solution.

own device. And this system is completely dynamic - maybe at some point only one user is presenting his work, but as soon as another decides to intervene and get up to expose his ideas, the visualization automatically adapts to this change. Even if someone is completely distracted, it is easy to get back in the discussion because everything is highlighted automatically and in real time.

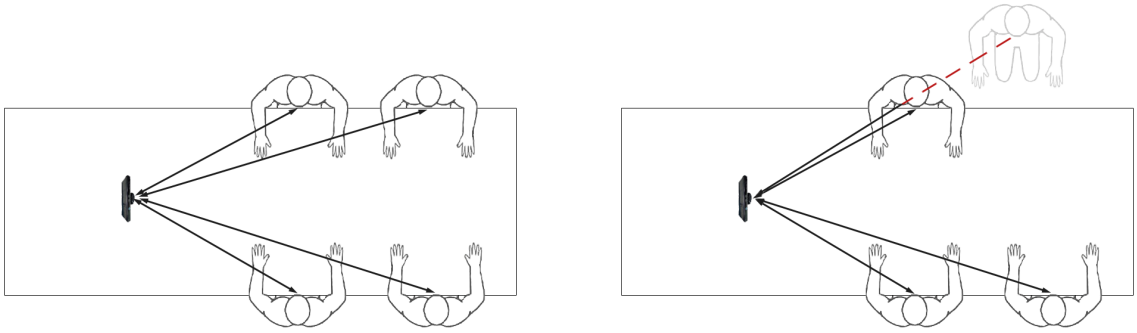
As for decision making, it benefits from our solution because subjects spend less effort interacting with the system - they are simply discussing like they would do without any software and decisions arise naturally. The process is never interrupted by peoples' explicit interaction with computers like plugging and unplugging cables for projecting information or even pressing buttons or adjusting sliders. Users can focus on their task and the system is there only to give them visual cues on what is going on - it never gets in their way.

A more in depth discussion of the impact of these proxemic interactions is available in section 5, where we compare the usage of our system with and without spatial knowledge. In both cases users can take advantage of the functionality discussed in this section - they can highlight their contributions. The difference is only in how this is done.

Without proxemics users must explicitly tell the system when this is supposed to happen. This is done by accessing the screen shown in Figure 3.11. This screen has a slider which controls the "relevance" value, which is what defines the size of the contributions of the user on screen. Each user can see the relevance of other users and can edit his own relevance. The mediator can edit everyone's relevance and can also use this screen to change the mediator and to choose a person's weight contributions.

With proxemics the relevance value is assigned automatically according to user distance to the shared screen. To add this feature we considered using the Proximity Toolkit developed by Nicolai Marquardt Greenberg et al. [12], but because the required detection was simple we took advantage of one of the modules already existent in YVision (a software developed by YDreams), which handled the complexity of dealing with a Kinect sensor. In practice, we just needed to connect the depth of each skeleton to the relevance value of the correspondent user.

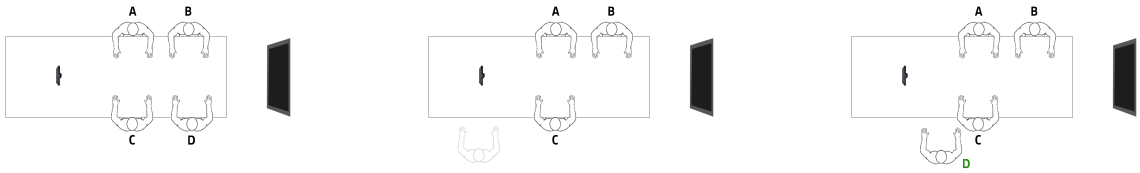
To handle the user identification we added a screen in the login phase of the system. In this screen there is a top-down view of the room, including a representation of the table, the screen, the sensor and the skeleton of each user as detected by the Kinect. The users are asked to tap the skeleton corresponding to them and the session can only begin once every user has done so.



(a) Scenario A - the sensor is able to detect every user correctly.

(b) Scenario B - one of the users is occluded by other, which makes it impossible for the sensor to detect him.

Figure 3.12: Occlusion of users by other users.



(a) Initial situation.

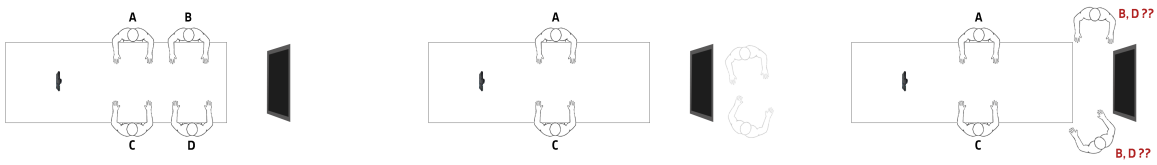
(b) User D stops being tracked.

(c) A new skeleton is detected by the sensor, it can only be user D.

Figure 3.13: Correct recovery from a situation with a missing skeleton.

3.5 Skeleton Detection

It is important to note that the skeleton detection mentioned in the previous section is not completely reliable, mainly due to occlusions. This is particularly true in this scenario, where users often sit side by side. As illustrated by Figure 3.12, the default position is supported by the sensor but a slight deviation is enough to occlude a user. The system is able to tolerate at most one missing user at any given time (Figure 3.13). If at any moment more than one user is occluded, then the user detection becomes unreliable (Figure 3.14). This happens because if only one user is missing and a new skeleton becomes available it's reasonable to assume that this skeleton belongs to the missing user. However, when there are two missing users and a new skeleton is detected there is no way to tell if it belongs to one user or the other without applying other techniques.



(a) Initial situation.

(b) Users B and D stop being tracked.

(c) Two unidentified skeletons are detected. There is no way to tell which one of them belongs to user B or D.

Figure 3.14: Situation where user detection recovery is not possible without other techniques.

In the test sessions with users, the disposition of the room was carefully thought to minimize the number of situations where this could happen. There was intentionally no room for the users to walk behind the shared screen and a big table discouraged them from walking behind the Kinect sensor, which was placed in the middle of the table, facing the screen. It was decided to have the Kinect attached to the table instead of the screen because the sensor is unable to function correctly at a distance smaller than 0.4 meters (using Near Mode). If the sensor was attached to the screen, then users at a distance smaller than 0.4 meters wouldn't be detected, which would go against one of the main focuses of the application - allowing users to present their contributions in the shared screen.

3.6 Summary

In this chapter we first discussed the problems of CSCW that are good candidates for being solved through the use of proxemics. We then explained our scenario in more detail, a process which was already well defined before we began our work. To support this scenario we developed a base system which can optionally be equipped with a module for detecting proxemics in the room. Lastly, we explained the main issues found when implementing our approach to the detection of moments when users wish to present their work.

In the following chapter we detail the development process, explaining the decisions that led to our solution.

Chapter 4

Development Process

The only way to guarantee that the final results are valid is by making sure that the system is built in accordance with the users' needs. The goal is to test a system that final users would actually use in their work, otherwise evaluating productivity and work results would become meaningless.

To meet all of those needs and expectations, we built our system using user-centered development guidelines. This way we also guarantee that we are not building what we think is most appropriate, but what users tell us is the most appropriate. In this chapter we describe that process, beginning in the identification of users, moving on to prototype development and finally integrating proxemics in the solution.

4.1 Identifying Stakeholders

For identifying stakeholders, we used the definition by Sharp et al. [34]. We identified the base stakeholders and proceeded to identify the satellite, supplier and client stakeholders based on those (Table 4.1).

The main stakeholders, the end-users, are mostly people working for water distribution companies. However, it was pointed out by the partners that in these meetings other people external to the company might be present - policy makers (mayors and/or deputies) and people from other organizations, for instance environmental or local organizations.

The development process was carried out at YDreams, with a valuable contribution from the partners, mainly LNEC, which was the main source for determining the usage scenario and for comments throughout the development process.

Asides from the end-users, it was also taken into consideration previous experience with the review board from the TRUST project. A correct understanding of the project as a whole was fundamental for the adequate development of the work package. We have also considered the directives of the mobile platforms we are targeting in order to assure that our software can be published.

In the case of this software, the managers are again the people in charge of the bigger project and the European Union as the commissioner of it all.

The scientific community is an important satellite stakeholder, as one of the goals is experimenting

End-Users	Water Professionals
	Policy makers
	Stakeholder groups and related organizations
Development Team	YDreams
	Partners: LNEC, IST, Addition, SINTEF
Legislators	Work Area review board
	App review board (iOS, Android, Windows Phone)
Managers	TRUST Management
	European Union
Satellites	Water service clients
	Scientific community
	Every European Union citizen

Table 4.1: Stakeholders identification.

with human-computer interaction, which yields important information. Asides from that, the software will be used by people who can have an impact on water service clients and, by consequence, every citizen of the European Union.

4.2 Defining User Profiles

Focusing on the end-users group of stakeholders and taking into consideration information provided by the partners, it is clear that the software must support three distinct groups: technical staff, management staff and external brainstorming advisors.

Technical staff is the group of people who have deep knowledge about the technical details discussed during the meeting. However, they tend to know less about the strategy and plan of the company.

Management staff has complementary knowledge in regards to the technical staff: they know more about the strategy and goals of the company but less about how to implement them at a low technical level.

The external brainstorming advisors are not required in the meeting, but are very common. They are people who don't need to have a great knowledge about the company, but are experienced in the meeting process. When present, they tend to take the lead.

The main difference between all these groups is that they have access to different information. Therefore, the software must be built in a way that makes it easy for any of the users to share and explain his point of view or ideas. The third group is a particularly important one, as it indicates that these meetings are mediated, i.e. there is a user who is in charge of leading the session and making sure that every user is able to contribute.

It is also important to consider that like in any meeting, users have different degrees of willingness to participate or engage in the session. This can be handled by the mediator in part but this is an indication that the software should either encourage participation, provide better tools to the mediator, or both.

4.3 Defining the User Group

For deciding who can provide valid answers in questionnaires and valid user input in tests, we identified the user group. Initially we wanted to target users only at the companies involved in the project, but since that would allow only a small number of replies, the user group was adapted to a more abstract one.

This is reasonable because the main focus of the session is not actually limited to water distribution companies. Defining objectives, criteria and metrics is something that can be done in the context of any company, organization or even work group as long as the methodology is explained beforehand. Given this realization, these users were considered:

Our user group is comprised of people aged over 18 from both genders. They are familiar with technology, used to working with computers, tablets and phones. They are used to be a part of a team.

As for the details regarding people from the TRUST project, input could be gathered together with our partners, allowing for more specific requirements to also be elicited.

4.4 Identifying Requirements

The main source for the requirements were the meetings with the partners in LNEC, who have a deep knowledge on what happens during the meetings the software must support. In addition to this, we asked a group of people inside YDreams and in other companies to answer a quick questionnaire (Appendix A) regarding the definition of objectives, criteria and metrics. The main goal of this questionnaire was deciding on how to present information. We gathered input from 33 different users - 16 of them had previous contact with the methodology. A summary of the results is available in Appendix B.

The requirements were then refined using several prototype iterations (discussed in more detail in the following section). The final list of requirements is available in Tables 4.2, 4.3, 4.4, 4.5 and 4.6.

Requirement	**Source**
Write down the mission of the organization	Partners
Write down the vision of the organization	Partners
Add objectives to a dimension (one of Social, Environment, Economic, Assets or Governance)	Partners
Add criteria to an objective	Partners
Add metrics to a criterion	Partners
Edit an item (objective, criteria or metric)	Partners
Remove an item	Partners
Change the order of an item	Questionnaire
Propose an item (before adding it to the current result)	Questionnaire
Propose an alternative to a proposed item	Questionnaire
Vote on alternative proposals	Questionnaire
Add objectives from the TRUST guidelines	Partners
Add metrics from a metrics library	Questionnaire
Assign a weight to a metric (one of 0.5, 0.75, 1, 1.5 or 2)	Questionnaire and Partners
Create a personal weight distribution chart	Questionnaire

Table 4.2: Functional Requirements.

Requirement	Source
Import metrics libraries using the same file format as AWARE-P software	Questionnaire
Export objectives, criteria, metrics and weight distribution defined in a session to a file	Partners
Import objectives, criteria, metrics and weight distribution from a file	Partners

Table 4.3: Data Requirements.

Requirement	Source
The system will run in a meeting room with a table and a shared screen	Partners
The shared screen will be running the server application	Partners
Each user can use his own device (personal computer or mobile devices) with a client application	Partners

Table 4.4: Environmental Requirements.

Requirement	Source
There is a special user, the mediator, who is in charge of leading the session	Partners
The normal users can visualize information and can only interact with the system when proposing items, voting, assigning weights to their personal weight distribution chart or changing their ”relevance” value	Partners

Table 4.5: User Requirements.

Requirement	Source
Visualize objectives, criteria and metrics in a tree-like horizontal structure, objectives on the left and metrics on the right	Questionnaire and Partners
Visualize objectives, criteria and metrics in a pie-chart structure, objectives in the center and metrics in the edges	Partners

Table 4.6: Usability Requirements.

4.5 Prototyping

In order to build our system incrementally with our users' input, we have iterated the following process for a few times: develop several low fidelity prototypes (see Figure 4.1); test the prototypes with experts at YDreams (due to lack of users' availability); present the most recent version of the prototypes to the partners and gather their opinion; integrate feedback on the prototypes and restart from the first step.

It was because of prototyping that we were able to refine most of the features implemented in the final version. We benefited from this method as we continuously experimented with ideas and understood that some of what we thought would be a good solution was not in fact practical. The most prominent example happened when we suggested using a circular representation for the set of objectives, criteria and metrics. We realized that it would be difficult to read the text in that representation, but it made it easier to see the weight distribution. For that reason we have opted to use a table visualization when

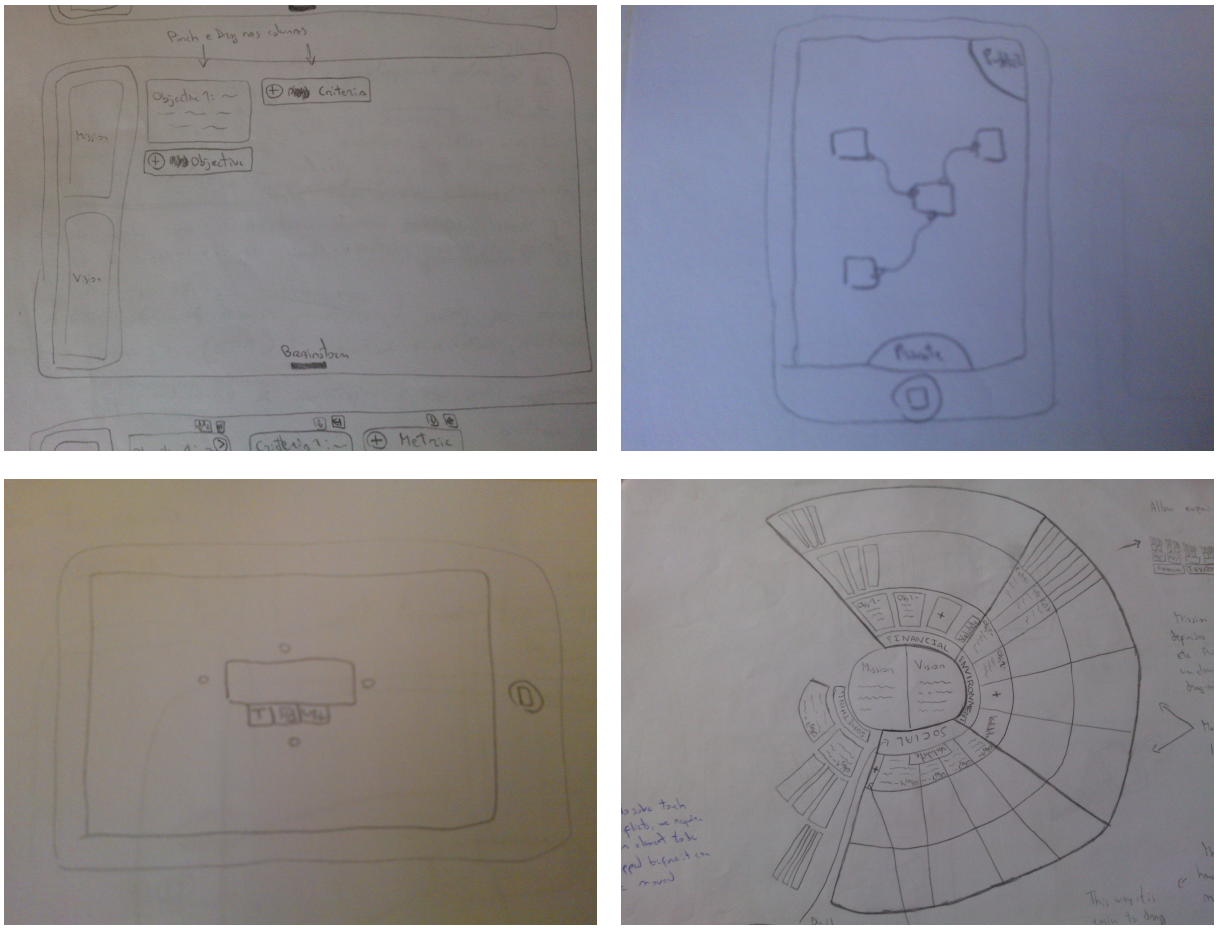


Figure 4.1: Low-fidelity prototypes.

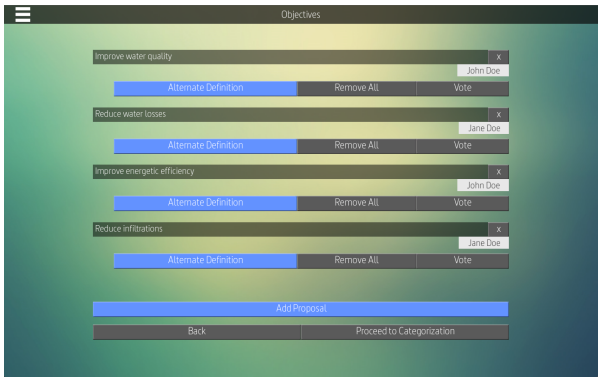


Figure 4.2: Users can defend their work in the proposals screen.

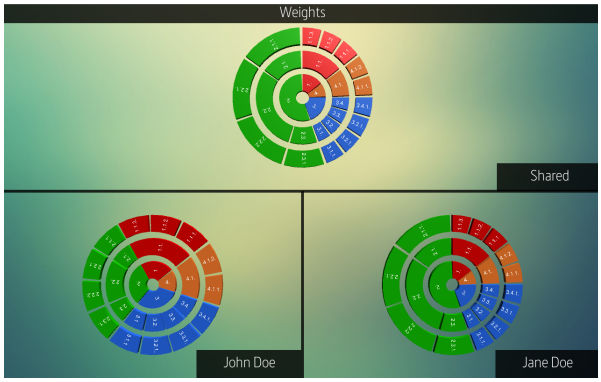


Figure 4.3: Users can present their personal weight distribution.

writing and the circle representation when assigning weights.

It was also through this process that we got a good knowledge of our users' needs, which allowed us to define the software architecture previously presented in section 3.3.

4.6 Wizard of Oz Testing

We considered that two functionalities would be good candidates for the implementation of proxemic interactions: mediator definition and argument presentation.

Dominance or leadership is one of the messages that can be conveyed through spatial communication. As such, we considered the mediator definition to be a good candidate - we could take advantage of this spatial data and automatically set the mediator.

There are two moments during the session when the users feel the need to defend their arguments: when defining objectives, criteria and metrics (Figure 4.2) and when defining the weight distribution (Figure 4.3). For these moments, the system could automatically adapt and change the visualization according to who is currently defending his argument.

Both features are supported by the system even without proxemics - the mediator can set the next mediator and all users have access to a screen where they can assign a value of "relevance" to a user. This value is directly proportional to the size of a user's contribution on the screen. Considering these two features, several alternatives were imagined with the same group of experts as before (Tables 4.7

Screen	The mediator is the person closest to the screen
Standing	The mediator is the last person who stood up
Screen & Standing	The mediator is the person standing closest to the screen
Arm	The mediator is the person who last raised his arm
Pointing	The current mediator points to the next mediator
Standard	The current mediator selects the next mediator using the software

Table 4.7: Different alternatives for usage of proxemics in mediator definition.

Screen	Relevance is proportional to distance to shared screen
Screen & Center	Same as 'Screen' but user must be facing the center of the room
Height	Relevance is proportional to the height of the body
Standard	Each user is able to change his relevance value using a slider in their device

Table 4.8: Different alternatives for usage of proxemics in argument presentation.

and 4.8).

Implementing all these alternatives would require a tremendous development effort, therefore we decided to test them using the Wizard of Oz method suggested by Kelley [20] - having users believe that the system was automatically detecting their position and movement, when in reality it was a developer who was controlling everything. Because no features were actually developed before testing, this method allowed us to explore a large amount of alternatives with no development effort.

4.6.1 Preparation

We invited a total of thirty users to test our alternatives. They were divided in groups of three, allowing us to carry ten sessions. These users were mostly software engineers (66%) and most of them had a small amount of experience at the company - 86% of the users had been at YDreams for less than a year. Further detail on the test population is available in Appendix E.

To achieve the illusion of a working system, a room was prepared with a Kinect device (which was turned on but not actually doing anything). Two users used a tablet each and the third user was given a laptop.

Each session took approximately one hour and thirty minutes and it had two parts, one for testing the mediator definition and another for testing the argument presentation. In the beginning of the session we did a brief (five minutes) presentation of the software and process, which was similar to the one used in the final tests, but explaining only the necessary features for these tests.

The Mediator Definition part began with handing out a script to the users - we decided this was better because there is no guarantee in a real session that the mediator will change. Therefore, we gave the users simple instructions:

Once you become the mediator, select a screen from the menu and say out loud the word "done". If the current mediator definition method is triggered by the current mediator (pointing and standard),

select the person on your right to be the next mediator. If the current mediator definition method is triggered by the users, wait for the person on your right to become the mediator. Repeat this process until the first mediator becomes mediator again.

Each group followed these instructions a total of six times, one for each method, taking approximately thirty minutes. The first mediator was chosen randomly and the sequence of methods was also changed randomly for every session.

Once all methods had been tested, we asked the users to fill the questionnaire available in Appendix C, which was targeted at understanding which method was the best in their opinion.

For the Argument Presentation part we decided to limit the tests only to the presentation of arguments in the objective definition phase - it is the easier one to understand because it is the first to take place and it provides a good sample for all the remaining moments of discussion, because they are all similar.

We gave users a scenario they were familiar with, which was also used in the final tests:

Suppose that the group is a team responsible for creating a new YDreams spin-out, YWear, whose focus is to create wearable technology. In this session you are going to list and prioritize the goals to achieve in the next six months, assuming the spin-out is starting now.

We also gave them a set of 20 objectives that we prepared with the partners. The idea was that the users would choose one objective and defend it. As a group they should then decide which objective was the most important and discard the others. By giving these objectives to the users, we were able to make our tests shorter in time because there was no need for users to come up with their own ideas, they could focus only on what was important for our tests: the discussion.

Each group did a total of four discussions, one for each argument definition method. Once an objective had been used by anyone, no one else could use it again. We allowed users to freely decide whether they would present their argument one at a time, two at a time, or even three at a time. We only set a maximum time limit of ten minutes. The order at which each method was tested was again randomized in different sessions.

For understanding the opinion of the users in regards to the tested argument definition methods, we gave them five minutes to fill a small questionnaire (available in Appendix D).

In order to make sure that there was no influence of the order in the results, we changed the order of the parts for each session: half began with the mediator definition tests and the remaining ones began with the argument presentation tests. Because the session required some time from the users, we also did a small five minute break in between parts.

4.6.2 Results

During the mediator definition method tests it was clear that users weren't feeling comfortable with a proxemics alternative to the standard solution. We noticed that some users weren't comfortable with the

pointing alternative, as it could be seen as rude. Also, having the person closest to the screen as the mediator, even when that person is not standing complicated the session in most cases, as users were more focused in not getting close to the screen than in contributing. This inadequacy was even clearer as soon as the results of the questionnaire were analyzed (full results in Appendix E).

The most popular proxemics alternative was the ”Screen + Standing” one. Yet, only 6 users (20%) considered it useful, only 2 users (7%) stated they would use it often and again only 2 users stated it was very easy to use. Even though the majority of the users (21 users, 70%) stated that they considered it more important for the users to gain control of the session, they still preferred the ”standard” method, which gives complete control to the mediator, over proxemics alternatives. The majority of the users considered all of the proxemics alternatives to be impractical. For every alternative, the number of users who stated they would never use it was always above 20 (67%), with a similar result for ease of use (very difficult to use was selected by more than 20 users in every proxemics alternative). It was unanimous that the best option was simply the standard one.

Regarding the argument presentation functionality, users were more receptive to the proxemics solutions. During the tests we observed that there were two alternatives that users were clearly appreciating: ”Screen” and ”Screen & Center”. This observation was later visible in the results (full results in Appendix F).

When asked if they would use each of the alternatives, there were actually more users replying they would use the Screen alternative (93%) or the Screen + Center (60%) alternative than those stating they would use the Standard alternative (43%). A similar number of users considered the same two alternatives to be easier to use than the standard one. It is relevant to note that users did consider the standard alternative to react better to their expectations, but even so they preferred using the Screen and Screen & Center alternatives (as supported by data discussed in the two previous items). This is an indicator that even though users felt they had a more accurate control using the standard alternative, the flow of the discussion benefited from using proxemics.

Even though the Height alternative was understood by 96% of the users, they didn’t consider it to be a good approach, as supported by the ease of use rating of this alternative - 80% considered it very difficult to use. Observation indicates that this happened because it was common for users to get up and unintentionally triggering a surge in their relevance.

Overall, the Screen alternative performed better than the Screen + Center alternative. This is supported by the questionnaire results, but could be understood via observation. What happened often was that users were presenting their argument, facing the center of the session, but as soon as they would turn slightly, even if it was to show a detail on the screen, their relevance dropped, which was against what they would want from the system - while they are presenting, the users wanted the system to highlight their work, regardless of the fact that they are facing the center. This observation seems to contradict F-Formations, but we may consider that the fact that there is a screen with relevant information makes that screen part of the formation and, as a consequence, there is no need for the user who is presenting to be facing the audience only.

4.7 Including Proxemics

The results from the previous stage allowed us to draw two important conclusions: there was no suitable alternative to the mediator definition method, apart from the standard option of allowing the users to press a button to select the next mediator; and users were very pleased with the possibility of a system automatically detecting when they were presenting their arguments. It reacted when and how they expected it to and as a result the feature was used more often than in the standard solution.

Considering that none of the mediator selection methods was well received except the standard one, we decided to implement none of the proxemics methods in this stage. Having a feature that was achieved through proxemics just for the sake of further exercising that option would certainly have a negative impact on the final results of this thesis - users could consider the proxemics solution to be more difficult to use because of this feature only, and not because of the overall experience with the system.

As for the argument presentation method, we decided to implement the most promising option according to the test results - the Screen alternative.

The integration of proxemics was done in the exact same project as the previous solution, which was initially complex because we needed to support several platforms. As an example, there was an obvious conflict in having Windows-dependent code handling the Kinect sensor in the same project that would be deployed to an iPad. However, having everything in the same project allowed for a much faster development process. To achieve this, we developed scripts that would ”hide” a part of the code depending on the platform and that would give different options to the users. When running the server application on a Windows computer, it is possible to select ”Use Proxemics”, which requires the Kinect sensor. However, this option is not available on any other OS. As for the client application, it can make use of the proxemics even without being connected to a Kinect sensor - all the detection is handled by the server and the data is sent through the network, guaranteeing that visualization is coherent throughout all the devices that are part of the session.

4.8 Summary

In this chapter we have presented how we took advantage of user-centered development guidelines to produce a system validated by users.

We first discussed who are the users the system must support by identifying stakeholders, user profiles and user group. Because of this work we were able to make the decision of including a broader range of users than what was initially planned. Considering only users from water companies would be ideal, but impractical and because the software is more general than that, we can still get reasonable results from other users - in this case, people working at YDreams.

Through a questionnaire and incremental prototypes we were able to define all of the system’s requirements and to build a solution according to those. This was achieved with minimal development effort, as it was based on creating prototypes with increasing fidelity according to user feedback.

Because understanding the best way of taking advantage of proxemics is not an easy task, we used the Wizard of Oz method, allowing us to test several alternatives.

We decided to leave one of the features implemented with proxemics out of the final prototype because results were not promising and focused on implementing only argument presentation. We were particularly careful in the way we integrated it with our solution, making sure it was easy to turn proxemics on or off.

This flexibility was fundamental during the evaluation stage, which is presented in the following chapter. Deciding whether we would use the proxemics solution or not simply required ticking a checkbox when starting the server. As for the client applications, absolutely nothing had to be done, they would automatically adapt to each of the solutions.

Chapter 5

Evaluation

For validating our hypothesis - that proxemics introduces a benefit in collaborative face-to-face meetings - we compared the two developed systems. Because we evaluated collaborative software, we consulted the Methodology for Evaluation of Collaborative Systems (Damianos et al. [6]), which considers four levels of evaluation: requirement, capability, service and technology.

According to the mentioned methodology, systems can be evaluated from a top-down perspective (requirement level to technology level) or from a bottom-up perspective. The intent is to be able to understand whether a system supports collaboration adequately at every level. If it doesn't, it can be excluded.

Our approach is similar - we perform an experiment with groups of users, giving them the same scenario and comparing metrics at different levels. This way it is possible to conclude which system is better at each level and overall.

Considering this framework, the difference between both systems we are evaluating is at the technological level - we have two different implementations of the same relevance attribution service - one requiring the user to change the relevance manually and another which detects the relevance automatically. However, this does not mean that we can only evaluate at the technology level, because other levels might be affected by the way people use the system. The only exception is the service level, since the service is exactly the same.

The framework describes types of tasks that collaborative systems support, of which we select the ones which are the focus of our system: Planning, Brainstorming and Decision Making. Considering the objectives of the session, these types are the most adequate - users are given a goal and have to deliver a plan for reaching that goal; they have to brainstorm in terms of objectives, criteria, metrics and weights; and they must reach several decisions throughout the session, choosing between alternative objectives, criteria, metrics and weight assignments.

To assess these types of tasks, we use the same evaluation framework, which explicits which metrics are relevant for each different type. Because some are present in more than one type, we aggregate them at different levels.

At the requirement level we are interested in understanding whether the different methods satisfy the

requirements. This can be done by measuring task outcome, cost, user satisfaction and efficiency. Task outcome includes the number of generated artifacts, task completion and user ratings of product quality. Cost includes learning effort, number of turns, length of turns and total time. User satisfaction includes process satisfaction, outcome satisfaction, individual and group participation satisfaction. Efficiency includes the number of artifacts / time and user ratings about efficiency.

At the capability level we want to know which method allows a better communication between users. This is defined by the number of turns per participant, user ratings on quality of communication, getting floor control, getting the attention of other participants and ability to interrupt.

At the technology level we want to know which system performs better and how people react to it. This can be answered using standard user interface metrics - fundamental interactions, time and user satisfaction, which is also partly covered at the requirements and capability level.

With all the metrics selected, we moved on to define the experiment.

5.1 Experiment

The experiment involved eight groups of exactly three users - each one of the groups completed one session (either with or without proxemics). This number of users and groups was chosen in order to have the highest number of sessions possible - we needed at least three users to have a meaningful discussion and we wanted to have a reasonable number of sessions to compare.

Our first attempt was again to involve the users from the TRUST project but we weren't able to find enough subjects for a meaningful experiment. For solving that problem, we invited people working at YDreams to be our test subjects and were careful in making sure that the sessions were as equivalent as possible. Most of the users (58%) were interns at YDreams with ages ranging from 20 to 25. These interns all share a background in either software engineering (65%) or electrical engineering (35%). The remaining subjects were designers (17%), project managers (12.5%) and marketing specialists (12.5%). These professionals were generally more experienced, with ages ranging from 27 to 50. Out of all users, 87.5% were male and 12.5% were female. In regards to nationality, 22 of the 24 users were Portuguese, 1 was Spanish and 1 was French.

To make sure that the experiments were equivalent, we selected two people from the intern group and one from the most experienced group for each session. This way it was not only guaranteed that the sessions were similar, but also that there would be a natural leader - typically, workers at a senior level have more experience and are a good fit for the role of mediator. We were also careful to group people together with other people they normally work with, increasing the chances of a dynamic and productive session.

All the test sessions took place in the same room, which was previously prepared for the effect, in the same way that it was done for the Wizard of Oz tests - a single table and a shared screen. We only changed one thing - in order to be sure that we didn't affect users' expectations, we removed the Kinect sensor from the room for non-proxemics sessions. If the sensor were there, it could be argued that users expected to use it somehow during the session and were disappointed not to. There were three devices

available for the users - two tablets and one laptop. The only restriction we applied was that the mediator would always use the laptop.

Each session included: 10 minutes for a pre-session briefing - explaining the process and software using a slide presentation (there were two versions of this presentation, one explaining the usage of the software with proxemics and another without); 5 minutes for a pre-session questionnaire; 60 minutes to carry out the session - if at the end of the 60 minutes users were not yet finished, the session would be stopped anyway; and 10 minutes for a post-session questionnaire.

The pre-session questionnaire and the post-session questionnaire were used for three purposes - measuring cost, communication and user satisfaction. We prepared a quick test covering topics on the process of the session and on the software usage and we gave users the same questionnaire before the session and after the session. This way we were able to measure which system is easier to learn. The post-session questionnaire also included questions regarding user satisfaction and communication. These questionnaires are available in Appendices G and H.

The scenario that was given to the users was intentionally small in scope and also intentionally centered around a concept that every user (a YDreams' employee) could relate to:

Suppose that the group is a team responsible for creating a new YDreams spin-out, YWear, whose focus is to create wearable technology. In this session you are going to list and prioritize the goals to achieve in the next six months, assuming the spin-out is starting now.

A mediator was assigned in the beginning of the session by mutual agreement between the users. Because too many variables would change if the mediator changed (device in use, meeting dynamics, times, etc.) we removed the ability to change the mediator from the test applications. This way, there was one and only one mediator throughout each session.

The mission and vision were written before the sessions with a specialist and given to the users, since they are not supposed to be a part of the session, but previously established:

- Mission - to create products that can be worn and that empower users with technology.
- Vision - our vision is to provide the most significant and aesthetically pleasant wearable technology products in the market.

During the session, we collected a series of values, in order to assess the previously discussed metrics. We instrumented the code and made it so that the server application would generate a file at the end of the session with the values we needed. Table 5.1 sums up which values were collected for each metric. Some of them are useful for more than one metric, but they appear where they are most relevant. The fundamental interactions considered were all clicks, taps and drag start, excluding text typing.

For measuring task outcome, we verified whether one of the methods produced more artifacts than the other (number of proposals, alternatives, objectives, criteria and metrics) and whether any method

would influence the users individually to make more suggestions (number of proposals and number of alternatives per user).

In terms of cost comparison, we wanted to know if there were any differences in duration, not only the total duration of the session but also the duration of each of its segments. These values are also relevant for usability comparison.

The efficiency allows us to analyze the production of artifacts in regards to time. Knowing only that users produce more artifacts is not enough, we want to know if users produce more artifacts per unit of time. To achieve this we measured the proposals, alternatives, objectives, criteria, metrics and weight proposals per minute. We also measured the number of weight proposals per minute in split-screen, which may indicate whether the use of split-screen influenced the number of weight proposals.

For comparing communication we want to know whether either method produced more turns and whether users participated more. This is measured by comparing data collected automatically (the distribution of user contributions over time) with the replies from the user satisfaction questionnaires, which have questions specifically targeted at this topic.

Finally, we compared the number of fundamental interactions in both methods. This metric is the most important in this work - this is the value where we expect to see a difference which is the starting point for the discussion of the remaining differences in all the other metrics.

Because we wanted to see if there were more differences in communication and time values we have also measured four more values through observation: the total time spent discussing alternatives in objectives, criteria and metrics (when two or more users engaged in discussion); the total time spent discussing weight assignment alternatives (when two or more users engaged in discussion); the total time when visualization is not adapted properly to alternatives discussion (if users are discussing alternatives, then the visualization should highlight those alternatives); and the total time when visualization is not adapted properly to weight discussion (when users are discussing weight distribution, the alternatives should be visible). These times could be collected two at a time, because alternative discussion never happens at the same time as weight discussion, therefore they were easily collected by an observer with two stopwatches.

5.2 Results

We illustrate our final results with a series of figures that compare the average results for the proxemics and non-proxemics sessions. In those figures, we also include two more values: the minimum and the maximum. We also present our final results with a confidence value calculated using the t-test.

At the requirements level, the differences in completion and generated artifacts are inconclusive. However, there is a great difference in the perception of the results by the users (view Figure 5.1). The average result satisfaction in the proxemics solution is 3.75 out 4, while it is only 3.17 out of 4 for the non-proxemics solution (confidence: 98%). As for learning, there were no significant differences in the results apart from the self-assessment. Users in proxemics sessions were more confident in their understanding of the process - 3.92 versus 3.33 (confidence: 99%) - and of the software - 3.83 versus

Level	Metric	Value
Requirement	Task Outcome	Total number of proposals
		Total number of alternatives
		Total number of objectives
		Total number of criteria
		Total number of metrics
		Number of proposals per user
		Number of alternatives per user
	Cost	Total session time
		Total objectives definition time
		Total criteria definition time
		Total metrics definition time
		Total time for weight assignment
	User Satisfaction	Covered in questionnaire
	Efficiency	Objectives per minute
		Criteria per minute
		Metrics per minute
		Proposals per minute
		Alternatives per minute
		Weight Proposals per minute
		Weight Proposals per split-screen minute
Capability	Communication	Map of proposal distribution for every user over time
		Map of alternative distribution for every user over time
		Number of users who showed weight distribution in split screen
Technology	User Interface	Number of fundamental interactions

Table 5.1: Values collected for each metric.

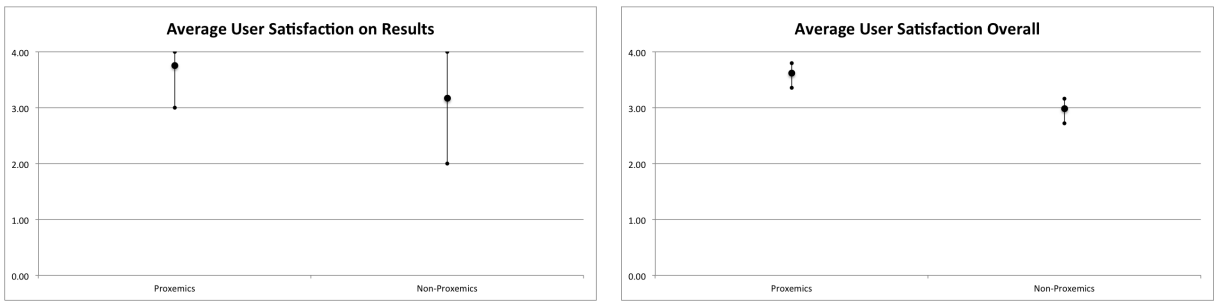


Figure 5.1: Comparison of user satisfaction with the results of the session and overall.

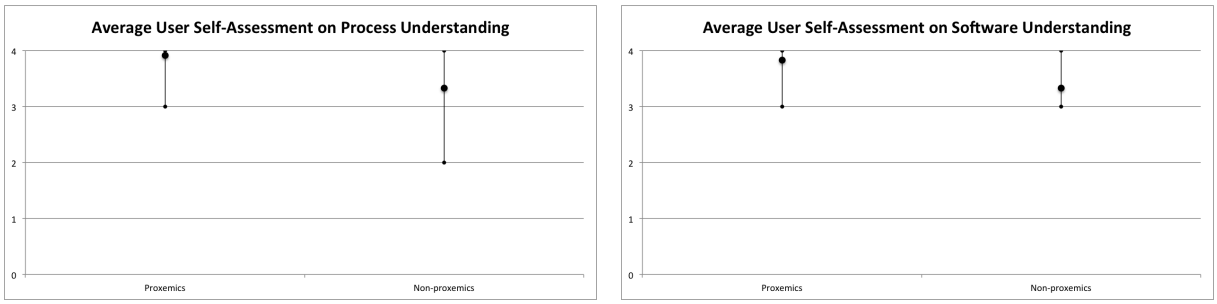


Figure 5.2: Comparison of self-assessment on process and software understanding.

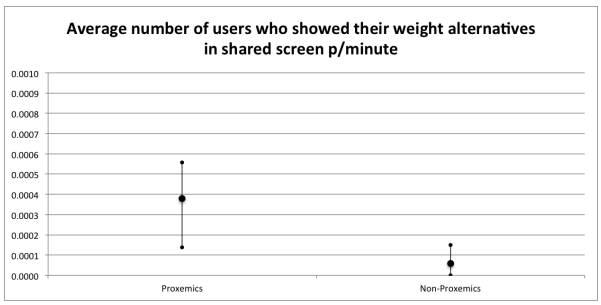


Figure 5.3: Comparison of the number of users who showed their weight assignment in the shared screen per minute.

3.33 (confidence: 98%). These results are shown in Figure 5.2. Finally, efficiency was also similar in most situations: objectives/minute, criteria/minute and metrics/minute. However, there were differences in the number of users who showed their weight alternatives in the shared screen per minute. In the proxemics sessions an average of 0.0004 users showed their weights, while only 0.0001 users did the same in the non-proxemics sessions (view Figure 5.3). This difference was calculated with a confidence of 97%.

For the capability level, we again point to the differences in the number of users who showed their weight assignment in the shared screen. By itself, this means only that there was more turn taking in this stage of the session. However, this fact acquires new relevance when analyzed together with the user satisfaction towards turn taking. Considering only the opinion of users about how they were able to express themselves and how others perceived them, we see that users in proxemics sessions were satisfied on average in a scale of 3.83 out of 4, while users in non-proxemics sessions were only satisfied in a scale of 3.28 out of 4 with a confidence of 99%.

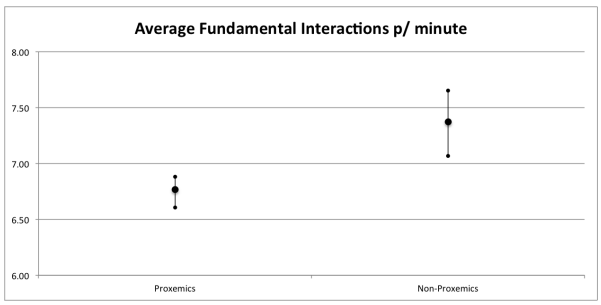


Figure 5.4: Comparison of the number of fundamental interactions.

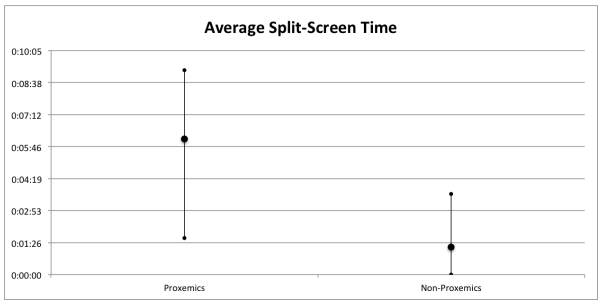


Figure 5.5: Comparison of the time spent in split screen.

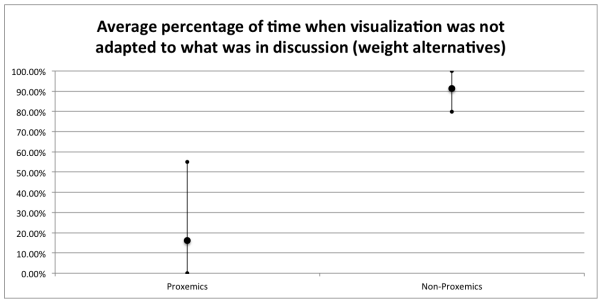


Figure 5.6: Comparison of the time when the visualization was not correctly adapted to the discussion of weight alternatives.

At the technology level, there is a statistical difference in the number of fundamental interactions. The average number of fundamental interactions p/minute was 6.77 for the proxemics solution and 7.37 for the non-proxemics solution with a confidence of 98% (view Figure 5.4). We have already discussed the user satisfaction with the outcome of the session and turn taking, but at the technology level we should evaluate the difference in overall user satisfaction (view Figure 5.1). The average satisfaction for a proxemics session is 3.62 out of 4, while for a non-proxemics session it is only 2.98 out of 4 (confidence: 99%). The differences in time are also mostly negligible. However, it is important to note that users in proxemic sessions spend an average of 4:52 minutes more in split screen than the users in non-proxemics sessions with a confidence of 95% (view Figure 5.5). Additionally, the amount of time when the visualization is not correctly adapted to the discussion of weight alternatives is much higher in the non-proxemics sessions - 91% of the time versus 16% of the time in the proxemics sessions, with a confidence of 99% (view Figure 5.6).

Due to the difficulty in finding test subjects, we needed to ask people who knew about our work to participate in the tests. Only five out of thirty subjects were familiar with the work, but still we made sure to isolate their responses and checked if they made any significant difference. We compared the results for user satisfaction and learning with and without these subjects and it showed no significant deviation. In 70% of the answers, the deviation was below 0.1 out of 4.0 points. The maximum deviation was only of 0.25 points, observed in user satisfaction in regards to visual adaptation to relevance in the proposal screen, which wouldn't change any of our conclusions. Given this fact, we decided to consider these users in our final analysis.

5.3 Discussion

Looking at the results we draw three important conclusions: 1. The proxemics solution yielded better user satisfaction at all levels; 2. Even though there weren't any significant differences in learning, users of the proxemics solution felt like they learned more; and 3. Users of the proxemics solution exercised more of the system, namely the visual adaptation of the shared screen to what is being discussed.

One of the reasons why users were more satisfied with the proxemics solution is the fact that it requires less fundamental interactions. This is a simple heuristic - if users can complete the same task with less work, they will be more satisfied with the interface. What's special about this case and about proxemics in general is that users can actually do more with less work. By correctly identifying when users would want the system to react, we have been able to produce a solution that reflects the users' needs without a conscious input and we have proved that, if applied correctly, this technique improves the users' perception of the system.

It might seem contradictory that in spite of having less fundamental interactions in the proxemics sessions, we could not find any significant difference in the duration of the sessions, which seems to indicate that fundamental interactions aren't that relevant. Because we've also seen an improvement in satisfaction, we believe that the reason for finding no difference in duration is simply because the dominant component of the session is discussion - users spend much more time discussing their proposals and alternatives than actually interacting with the system. This is supported by the low number of interactions in the sessions - between 6 and 7 interactions per minute -, and also by observing the users.

The improvement in satisfaction is also what explains why proxemics users felt like they learned more, even though they didn't. Users had a more positive attitude at the end of the session and felt better with themselves and with the results. The importance of this is clear: if users feel like they have a better understanding of the system and of the process, they will be more receptive to use it again. Because they are comfortable with it and because they feel like they can get good results with it, users will want to use the system again and to incorporate it in the company's work flow.

Knowing that users of the proxemics system took advantage of the adaptive visualization features makes us realize that this is something that users wanted, but that was not supported correctly in a non-proxemics solution. It could be argued that this could be because the non-proxemics solution needed to be refined, but that's where the whole development process comes into play. We are sure that we developed two systems that correspond to the users' needs because we developed and refined them with the users' help. In other words, this means that we have met a users' need through proxemics, that could not be achieved without proxemics.

In the case of this scenario, the fact that we have met more of the users' needs could potentially lead to higher adoption rates of the software and of the process, which is exactly the goal of a good CSCW application.

Chapter 6

Conclusions

We hypothesized that CSCW applications could benefit from the use of proxemics in its design and implementation, as it allows the reduction of interaction protocol and a bigger focus on the task. We targeted two problems - breakdown of intuitive decision making and awareness in regards to what is being done. To address these issues we developed a system for a particular scenario that takes advantage of the moments when people intend to present their work by automatically adapting the visualization to what is in discussion. For understanding the impact of this feature, we compared two versions of our system with the same functionalities but with a fundamental difference - one took advantage of proxemics and the other didn't. The results show that proxemics effectively reduces the interaction protocol and as a results users are more satisfied with the system and their work.

The major accomplishment of this thesis is understanding exactly where proxemics is responsible for an improvement and to conclude that it addresses the problems we identified earlier. We observed that with proxemics there are less fundamental interactions, which shows that there is a reduction of protocol, improving user satisfaction. We were able to achieve this conclusion because of a methodical comparison between a system with no proxemic interactions and a system with proxemic interactions implemented according to user feedback. This feedback is what allowed us to define exactly where and how we should include proxemics and it proved that users are receptive to proxemics even in a serious environment, as long as it is implemented carefully and not only as an added interaction method. The main evaluation was done in a final prototype and exercised the full capabilities of the software, which is something that is not usually found when evaluating proxemics. We didn't compare only how proxemics affected single interactions, but most importantly how it affected the whole system and how users perceived it.

As for individual interactions, this thesis also presents experiments on several methods for using proxemics data in mediator definition and information visualization. The most significant observation was that even though proxemics seems to be an obvious candidate for mediator definition, our tests revealed that it was not well received in our scenario. One possibility for why this happened is because spatial information is not enough to decide the mediator - maybe professional and relationships background also play an important role. Another option is that it is a system too complex for such a simple task -

using a simple button in the software solves the issue and that is enough in some situations. This subject could be further explored in the future, as well as the remaining experiments - it would be interesting to know if they fail or succeed in different scenarios and what are the reasons for that to happen.

We evaluated a single session but conjecture that proxemics would also have an impact in the long term: because users feel more comfortable with the system and are more confident about their work, the adoption rates would be higher in theory. In the future, it could be interesting to assess if this is in fact true by having different companies use the software for more than one session and reporting on user satisfaction and adoption over time. This is no simple task, however, as companies would have to be willing to incorporate both the process and the software in their environment, otherwise validity of the results could be questioned.

With better technology we could also think about improving the system itself. With more accurate hardware and algorithms in place we could think about taking further advantage of relationships between people and machines. The sort of interactions explored in Marquardt's case studies could eventually be explored without extra equipment that made using them in our scenario impractical. As an example, we could think about having users share information between personal devices only, without having it in the shared screen, allowing users to form work groups easily.

Reliability could also be improved by addressing problems with occlusions, which were previously discussed in section 3.4. This could be done by taking advantage of more advanced hardware - the newest version of the Kinect sensor, for instance, allows more accurate and faster detection, improving situations when users move too fast and the system takes some time to react. Another option would be using a method like the one suggested by Zhang et al. [36], who have used particle filtering and sampling to detect skeletons from a set of points in space obtained using multiple sensors.

Using proxemics in interaction still has a long way to go - until now it has mostly been used in interaction experiments, but we have shown the benefits of using it in a real world CSCW scenario. We were able to take advantage of a small part of all the spatial information communicated during a meeting and it already improved the users' opinion dramatically. With future technologies and experiments we can only expect to see much more improvements with the goal of achieving a level of interaction with machines closer to how we interact with each other.

Bibliography

[1] Helena Alegre and Sérgio T. Coelho. Infrastructure asset management of urban water systems. *Water Supply System Analysis - Selected Topics*, 2012.

[2] Michelle Annett, Tovi Grossman, Daniel Wigdor, and George Fitzmaurice. Medusa: A proximity-aware multi-touch tabletop. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 337–346, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1.

[3] Kevin Au and Mike Cheung. Intra-cultural variation and job autonomy in 42 countries. *Organization Studies*, 25(8):1339–1362, 2004.

[4] Jeremy Cooperstock, Koichiro Tanikoshi, Garry Beirne, Tracy Narine, and William A. S. Buxton. Evolution of a reactive environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pages 170–177, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-84705-1.

[5] Marco Cristani, Giulia Paggetti, Alessandro Vinciarelli, Loris Bazzani, Gloria Menegaz, and Vittorio Murino. Towards computational proxemics: Inferring social relations from interpersonal distances. In *Proceedings of IEEE International Conference on Social Computing*, pages 290–297, 2011.

[6] Laurie Damianos, Lynette Hirschman, Robyn Kozierok, Jeffrey Kurtz, Andrew Greenberg, Kimberley Walls, Sharon Laskowski, and Jean Scholtz. Evaluation for collaborative systems. *ACM Comput. Surv.*, 31(2es), June 1999. ISSN 0360-0300.

[7] Paul Dietz and Darren Leigh. Diamondtouch: a multi-user touch technology. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pages 219–226. ACM, 2001.

[8] Kurt Dohse, Thomas Dohse, Jeremiah Still, and Derrick Parkhurst. Enhancing multi-user interaction with multi-touch tabletop displays using hand tracking. In *ACHI*, pages 297–302. IEEE Computer Society, 2008.

[9] Paul Dourish and Victoria Bellotti. Awareness and coordination in shared workspaces. In *Proceedings of the 1992 ACM Conference on Computer-supported Cooperative Work*, CSCW '92, pages 107–114, New York, NY, USA, 1992. ACM. ISBN 0-89791-542-9.

[10] Scott Elrod, Richard Bruce, Rich Gold, David Goldberg, Frank Halasz, William Janssen, David Lee, Kim McCall, Elin Pedersen, Ken Pier, John Tang, and Brent Welch. Liveboard: A large interactive display supporting group meetings, presentations, and remote collaboration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pages 599–607, New York, NY, USA, 1992. ACM. ISBN 0-89791-513-5.

[11] Barry Gerhart and Meiyu Fang. National culture and human resource management: assumptions and evidence. *International Journal of HRM*, 16(6):971–986, 2005.

[12] Saul Greenberg, Nicolai Marquardt, Till Ballendat, Rob Diaz-Marino, and Miaosen Wang. Proxemic interactions: the new ubicomp? *interactions*, 18(1):42–50, January 2011. ISSN 1072-5520.

[13] Jonathan Grudin. Why cscw applications fail: Problems in the design and evaluationof organizational interfaces. In *Proceedings of the 1988 ACM Conference on Computer-supported Cooperative Work*, CSCW '88, pages 85–93, New York, NY, USA, 1988. ACM. ISBN 0-89791-282-9.

[14] Edward Hall. *The Hidden Dimension*. Anchor, 1966. ISBN 0385084765.

[15] Geert Hofstede. *Culture's consequences: International differences in work-related values*. Sage Publications, Beverly Hills, CA, 1980.

[16] William Hutama, Peng Song, Chi-Wing Fu, and Wooi Boon Goh. Distinguishing multiple smartphone interactions on a multi-touch wall display using tilt correlation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 3315–3318, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9.

[17] Mikkel Rønne Jakobsen, Y. Sahlemariam Haile, Søren Knudsen, and Kasper Anders Søren Hornbæk. Information visualization and proxemics: design opportunities and empirical findings. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2386–2395, 2013. ISSN 1077-2626.

[18] Robert Johansen. *GroupWare: Computer Support for Business Teams*. The Free Press, New York, NY, USA, 1988. ISBN 0029164915.

[19] Wendy Ju, Brian Lee, and Scott Klemmer. Range: exploring proxemics in collaborative whiteboard interaction. In Mary Beth Rosson and David J. Gilmore, editors, *CHI Extended Abstracts*, pages 2483–2488. ACM, 2007. ISBN 978-1-59593-642-4.

[20] John Kelley. An iterative design methodology for user-friendly natural language office information applications. *ACM Trans. Inf. Syst.*, 2(1):26–41, January 1984. ISSN 1046-8188.

[21] Adam Kendon. *Conducting Interaction: Patterns of Behavior in Focused Encounters*. Studies in interactional sociolinguistics. Cambridge University Press, 1990. ISBN 9780521380676.

[22] Meredith Ringel Morris, Danyel Fisher, and Daniel Wigdor. Search on surfaces: Exploring the potential of interactive tabletops for collaborative search tasks. *Inf. Process. Manage.*, 46(6):703–717, November 2010. ISSN 0306-4573.

[23] Gary Olson and Judith Olson. Distance matters. *Hum.-Comput. Interact.*, 15(2):139–178, September 2000. ISSN 0737-0024.

[24] Elin Rønby Pedersen, Kim McCall, Thomas Moran, and Frank Halasz. Tivoli: an electronic whiteboard for informal workgroup meetings. In Stacey Ashlund, Kevin Mullet, Austin Henderson, Erik Hollnagel, and Ted N. White, editors, *INTERCHI*, pages 391–398. ACM, 1993. ISBN 0-89791-574-7.

[25] Jun Rekimoto. Pick-and-drop: A direct manipulation technique for multiple computer environments. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, pages 31–39, New York, NY, USA, 1997. ACM. ISBN 0-89791-881-9.

[26] Jun Rekimoto. A multiple device approach for supporting whiteboard-based interactions. In *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 344–351, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-30987-4.

[27] Mahsan Rofouei, Andrew Wilson, A.J. Brush, and Stewart Tansley. Your phone or mine?: Fusing body, touch and device sensing for multi-user device-display interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 1915–1918, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4.

[28] Enrico Rukzio, Albrecht Schmidt, Heinrich Hussmann, and Media Informatics. An analysis of the usage of mobile phones for personalized interactions with ubiquitous public displays. workshop ubiquitous display environments, ubicomp 2004. In *Ubiquitous Public Displays. Workshop on Ubiquitous Display Environments, UbiComp 2004*, 2004.

[29] Edward Sapir. *Unconscious Patterning of Behavior in Society*. 1927.

[30] Dominik Schmidt, Fadi Chehimi, Enrico Rukzio, and Hans Gellersen. Phonetouch: A technique for direct phone interaction on surfaces. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 13–16, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0271-5.

[31] Johannes Schöning, Michael Rohs, and Antonio Krüger. Using Mobile Phones to Spontaneously Authenticate and Interact with Multi-touch Surfaces. In *PPD'08. Workshop on designing multi-touch interaction techniques for coupled public and private displays*, 2008.

[32] Stacey Scott, Sheelagh Carpendale, and Kori Inkpen. Territoriality in collaborative tabletop workspaces. In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*, CSCW '04, pages 294–303, New York, NY, USA, 2004. ACM. ISBN 1-58113-810-5.

[33] Khoovirajsingh Seewoonauth, Enrico Rukzio, Robert Hardy, and Paul Holleis. Touch & connect and touch & select: Interacting with a computer by touching it with a mobile phone. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, pages 36:1–36:9, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-281-8.

[34] Helen Sharp, Anthony Finkelstein, and Galal Galal. Stakeholder identification in the requirements engineering process. In *Proceedings of the 10th International Workshop on Database & Expert Systems Applications*, DEXA '99, pages 387–, Washington, DC, USA, 1999. IEEE Computer Society. ISBN 0-7695-0281-4.

[35] Robert Sommer. Personal Space. The Behavioral Basis of Design. *eric.ed.gov*, 1969.

[36] Licong Zhang, Jurgen Sturm, Daniel Cremers, and Dongheui Lee. Real-time human motion tracking using multiple depth cameras. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

Appendix A

IAM Questionnaire

Objective Definition

Consider a situation where you are tasked with defining a group of objectives for a project in your company. Each objective can be evaluated through a set of criteria and each criterion can be measured through a set of metrics. You can then assign different relative weights to each metric, which will allow you to understand which objectives are a priority.

*Required

1. **How old are you?** *
   Mark only one oval.
   - Less than 20
   - Between 20 and 30 (exclusive)
   - Between 30 and 40 (exclusive)
   - Between 40 and 50 (exclusive)
   - Between 50 and 60 (exclusive)
   - 60 or more

2. **What is your current job position?** *

   _____

3. **Do you have any experience with IAM (Infrastructure Asset Management)?** *
   Mark only one oval.
   - Yes
   - No

4. **Given a set of objectives, criteria and metrics, how would you organize them?** *
   Mark only one oval.
   - Vertically: objectives on the top, metrics on the bottom
   - Vertically: metrics on the top, objectives on the bottom
   - Horizontally: objectives on the left, metrics on the right
   - Horizontally: metrics on the left, objectives on the right
   - Other: _____

Figure A.1: Objective definition questionnaire, page 1.

5. **How would you sort each of these items?** *
Mark only one oval per row.

	Alphabetically	By importance	By insertion order	Other
Objectives	◯	◯	◯	◯
Criteria	◯	◯	◯	◯
Metrics	◯	◯	◯	◯

6. **If you selected "other" in one or more of the items in the previous question, which sorting method would you use instead?** *

.....

7. **What sources do you use when defining objectives, criteria and/or metrics?** *
Tick all that apply.

- ☐ Already existing guidelines from outside the company
- ☐ Already existing guidelines from inside the company
- ☐ Brainstorming with other members of the team
- ☐ Other: .....

8. **Suppose you are in a team meeting for defining objectives, criteria and metrics. Rate the following visualization options according to your preference (1 - low preference; 4 - high preference):** *
Mark only one oval per row.

	1	2	3	4
Only objectives	◯	◯	◯	◯
Only criteria	◯	◯	◯	◯
Only metrics	◯	◯	◯	◯
Objectives and criteria	◯	◯	◯	◯
Objectives and metrics	◯	◯	◯	◯
Criteria and metrics	◯	◯	◯	◯
Objectives, criteria and metrics	◯	◯	◯	◯

9. **In a team, how do you choose the weight to assign to each metric?** *
Mark only one oval.

- ◯ Work as a team on the same weight distribution
- ◯ Anyone can work on their own weight distribution and then present it
- ◯ Other: .....

Figure A.2: Objective definition questionnaire, page 2.

Appendix B

IAM Questionnaire Results

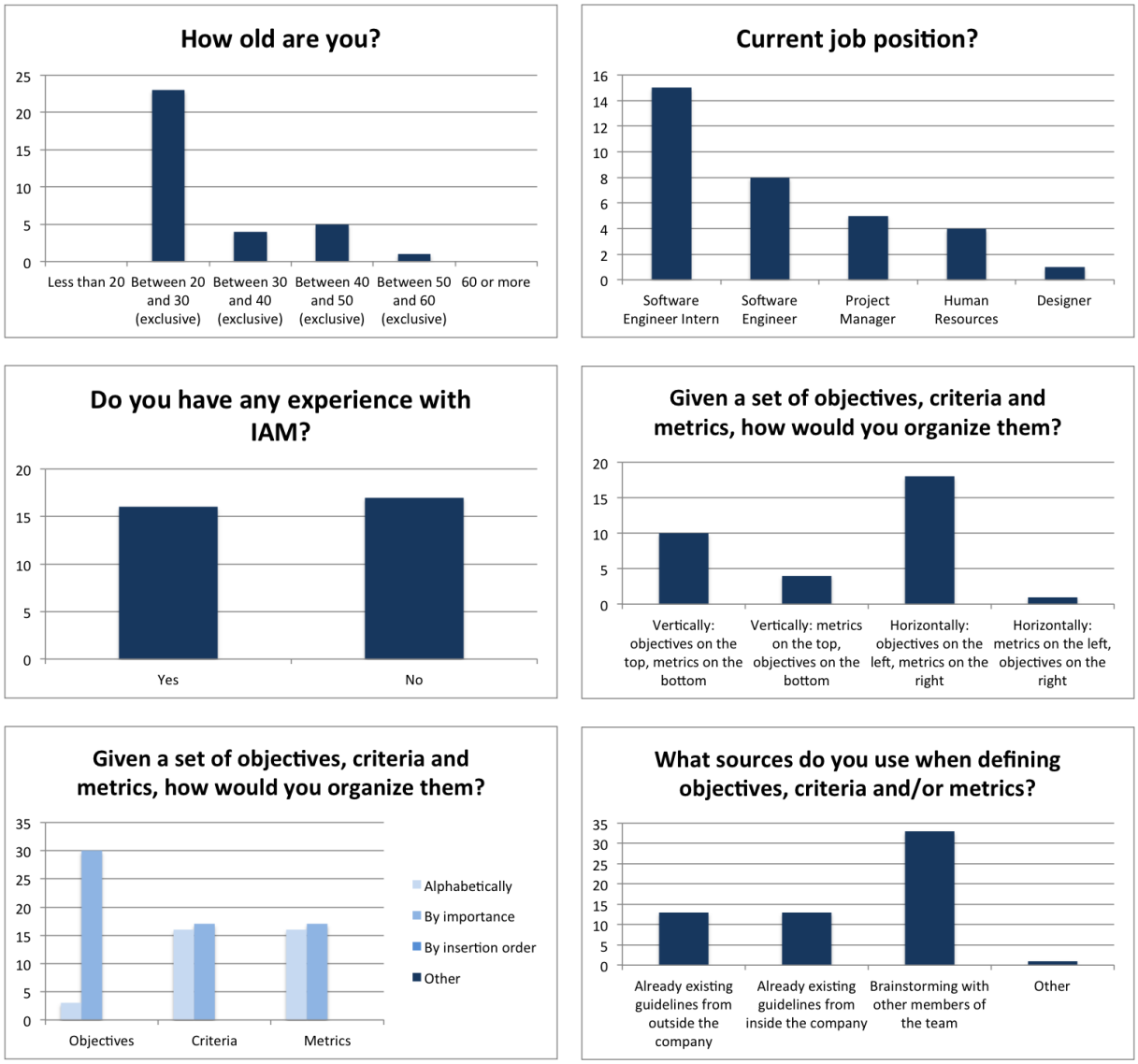
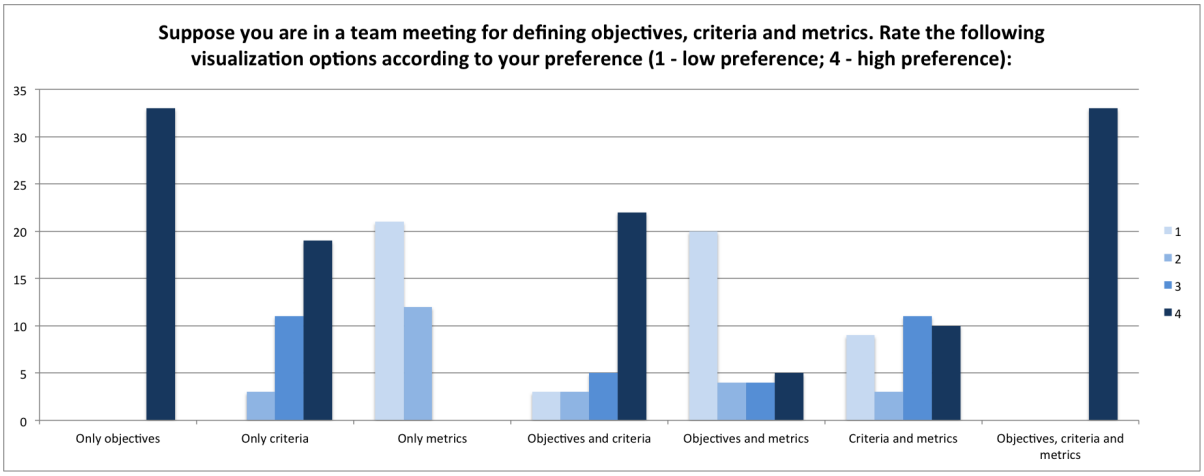


Figure B.1: IAM questionnaire results (1/2).



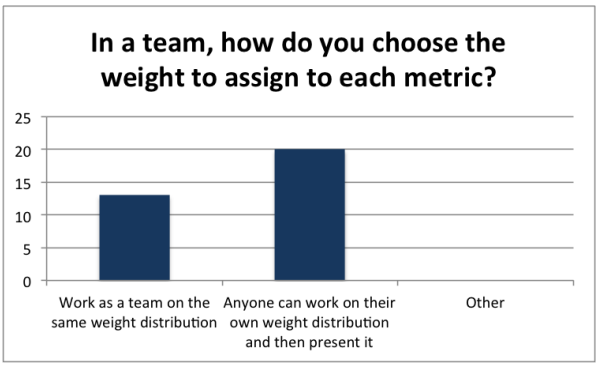


Figure B.2: IAM questionnaire results (2/2).

Appendix C

Mediator Definition Questionnaire

Mediator Definition Method

You have tried the following alternatives:
- Screen - the mediator is the person closest to the screen;
- Standing - the mediator is the last person who stood up;
- Screen + Standing - the mediator is the person standing closest to the screen;
- Arm - the mediator is the person who last raised his arm;
- Pointing - the current mediator must point to the person who he wants to be the next mediator;
- Normal - there is a screen where the current mediator can select the next mediator.

Considering your experience, please reply to the following questionnaire.

*Required

1. **Session number?** *

   ---

2. **What is your current position in your company?** *

   ---

3. **For how long have you been in this position?** *
   Mark only one oval.

   - Less than 6 months
   - Less than 1 year
   - Less than 3 years
   - More than 3 years

4. **Select the alternative where you felt that the current mediator has greater control over who the next mediator is:** *
   Mark only one oval.

   - Screen
   - Standing
   - Screen + Standing
   - Arm
   - Pointing
   - Normal

Figure C.1: Mediator definition questionnaire, page 1.

5. **Select the alternative where you felt that each user has a greater autonomy to become the mediator:** *

Mark only one oval.

- Screen
- Standing
- Screen + Standing
- Arm
- Pointing
- Normal

6. **Which do you think is more important in this kind of meetings?** *

Mark only one oval.

- Current mediator has the control over who the next mediator is
- Each user is able to become the mediator at any time

7. **For each alternative, select whether you consider it a gimmick or something that could be useful in this meeting scenario.** *

Mark only one oval per row.

	Gimmick	Useful
Screen		
Standing		
Screen + Standing		
Arm		
Pointing		
Normal		

8. **Select an option for each alternative you have tried.** *

Mark only one oval per row.

	1 - Would never use it	2	3	4 - Would use it often
Screen				
Standing				
Screen + Standing				
Arm				
Pointing				
Normal				

Figure C.2: Mediator definition questionnaire, page 2.

9. **Rate each alternative according to ease of use (consider speed, responsiveness and amount of work required form the user).** *

Mark only one oval per row.

	1 - Very difficult to use	2 - Difficult to use	3 - Easy to use	4 - Very easy to use
Screen				
Standing				
Screen + Standing				
Arm				
Pointing				
Normal				

10. **Please use this space to provide us with any further comment you might have (why you think an alternative doesn't work, other alternative suggestions, etc.).**

.....

.....

.....

.....

.....

Powered by Google Forms

Figure C.3: Mediator definition questionnaire, page 3.

Appendix D

Argument Presentation Questionnaire

Argument Presentation Method

You have tried the following alternatives:
- Screen - relevance is proportional to distance to shared screen;
- Screen & Center - relevance is proportional to distance to shared screen as long as the user is facing the center of the meeting;
- Height - relevance is proportional to the height of the body;
- Mediator - relevance is controlled by the mediator.

Considering your experience, please reply to the following questionnaire.

*Required

1. **Session number?** *

   _____

2. **What is your current position in your company?** *

   _____

3. **For how long have you been in this position?** *
   Mark only one oval.

   - ( ) Less than 6 months
   - ( ) Less than 1 year
   - ( ) Less than 3 years
   - ( ) More than 3 years

4. **Select an option for each alternative you have tried.** *
   Mark only one oval per row.

	1 - Would never use it	2	3	4 - Would use it often
Screen	( )	( )	( )	( )
Screen & Center	( )	( )	( )	( )
Height	( )	( )	( )	( )
Mediator	( )	( )	( )	( )

Figure D.1: Argument presentation questionnaire, page 1.

5. **Rate each alternative according to ease of use (consider speed, responsiveness and amount of work required form the user).** *
Mark only one oval per row.

	1 - Very difficult to use	2 - Difficult to use	3 - Easy to use	4 - Very easy to use
Screen				
Screen & Center				
Height				
Mediator				

Consider each sentence and alternative:

6. **The system reacted when and how I expected it to.** *
Mark only one oval per row.

	1 - totally disagree	2 - disagree	3 - agree	4 - totally agree
Screen				
Screen & Center				
Height				
Mediator				

7. **I understood the relationship between what I did and how the system reacted.** *
Mark only one oval per row.

	1 - totally disagree	2 - disagree	3 - agree	4 - totally agree
Screen				
Screen & Center				
Height				
Mediator				

8. **There were moments when the system reacted without my intention.** *
Mark only one oval per row.

	1 - totally disagree	2 - disagree	3 - agree	4 - totally agree
Screen				
Screen & Center				
Height				
Mediator				

Figure D.2: Argument presentation questionnaire, page 2.

9. **For each alternative, select whether you consider it a gimmick or something that could be useful in this meeting scenario.** *

Mark only one oval per row.

	Gimmick	Useful
Screen		
Screen & Center		
Height		
Mediator		

10. **Please use this space to provide us with any further comment you might have (why you think an alternative doesn't work, other alternative suggestions, etc.).**

.....

.....

.....

.....

.....

Powered by Google Forms

Figure D.3: Argument presentation questionnaire, page 3.

Appendix E

Mediator Definition Questionnaire Results

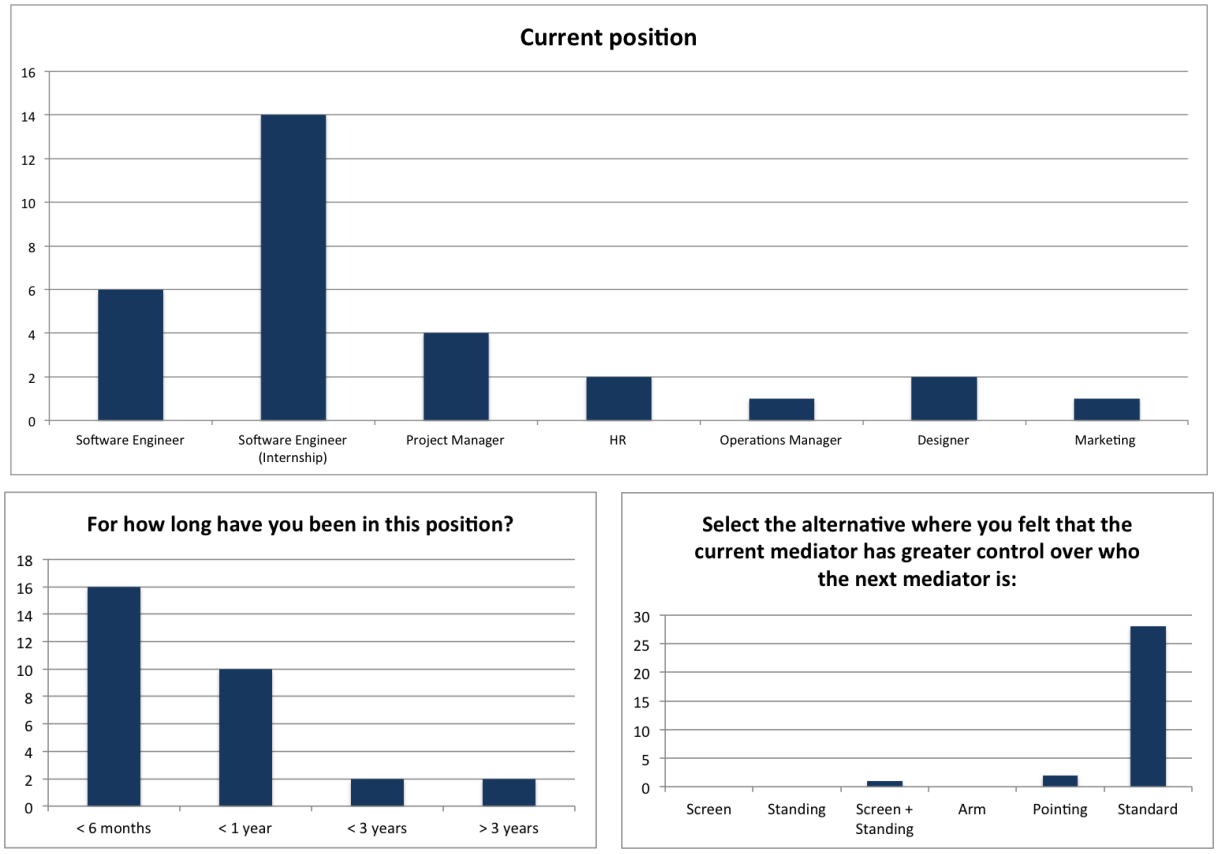


Figure E.1: Mediator definition questionnaire results (1/2).

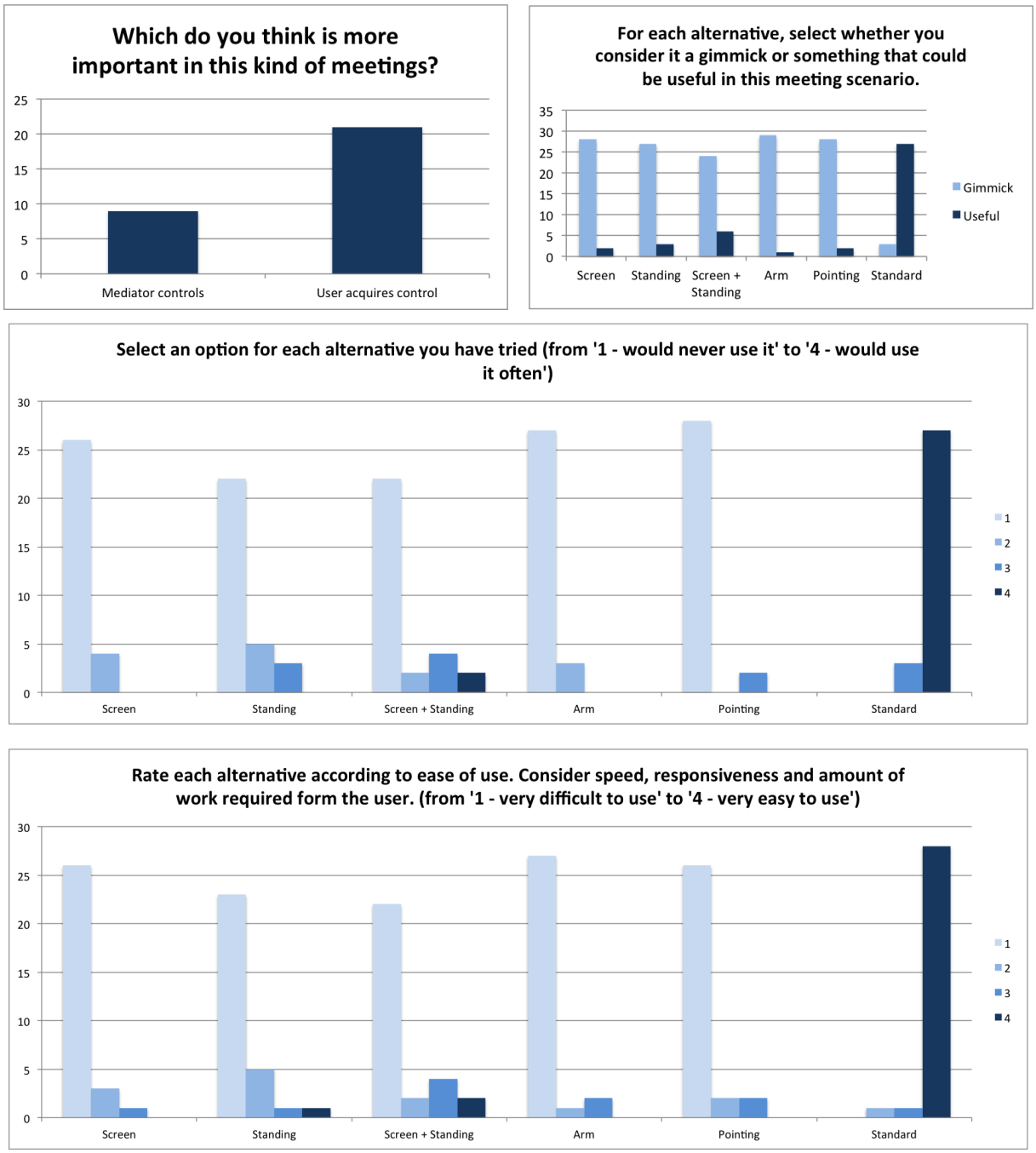
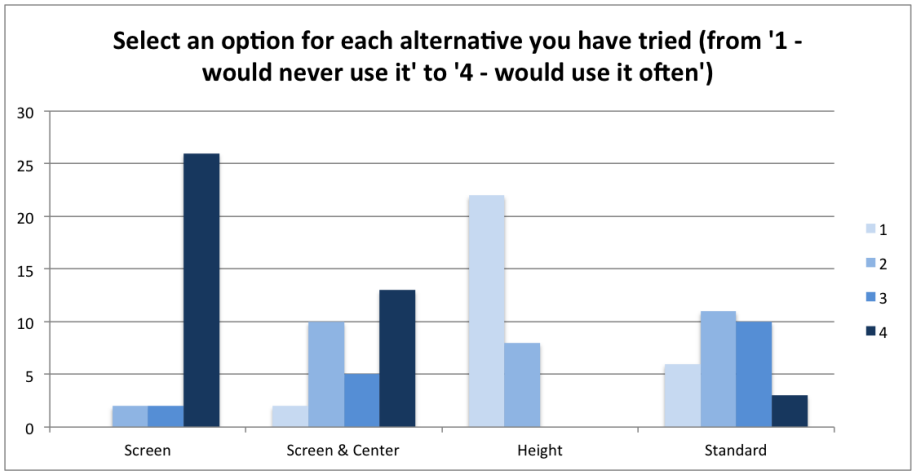


Figure E.2: Mediator definition questionnaire results (2/2).

Appendix F

Argument Presentation Questionnaire Results



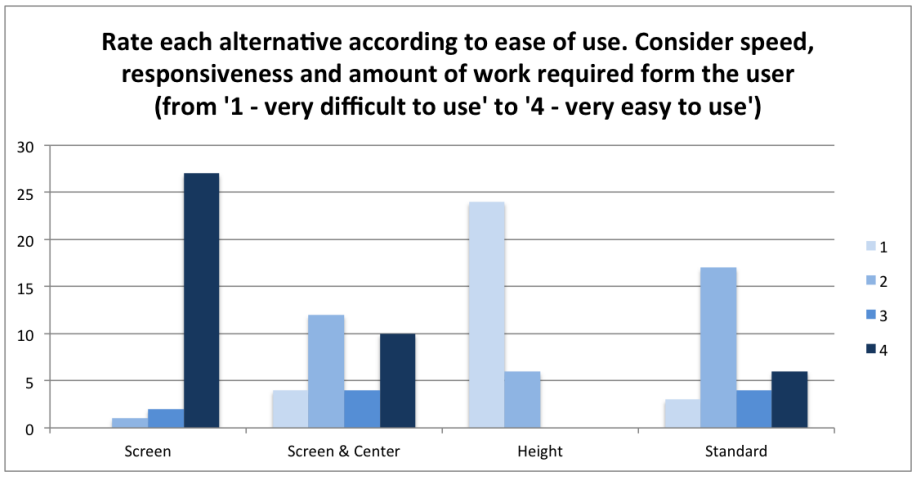


Figure F.1: Argument presentation questionnaire results (1/2).

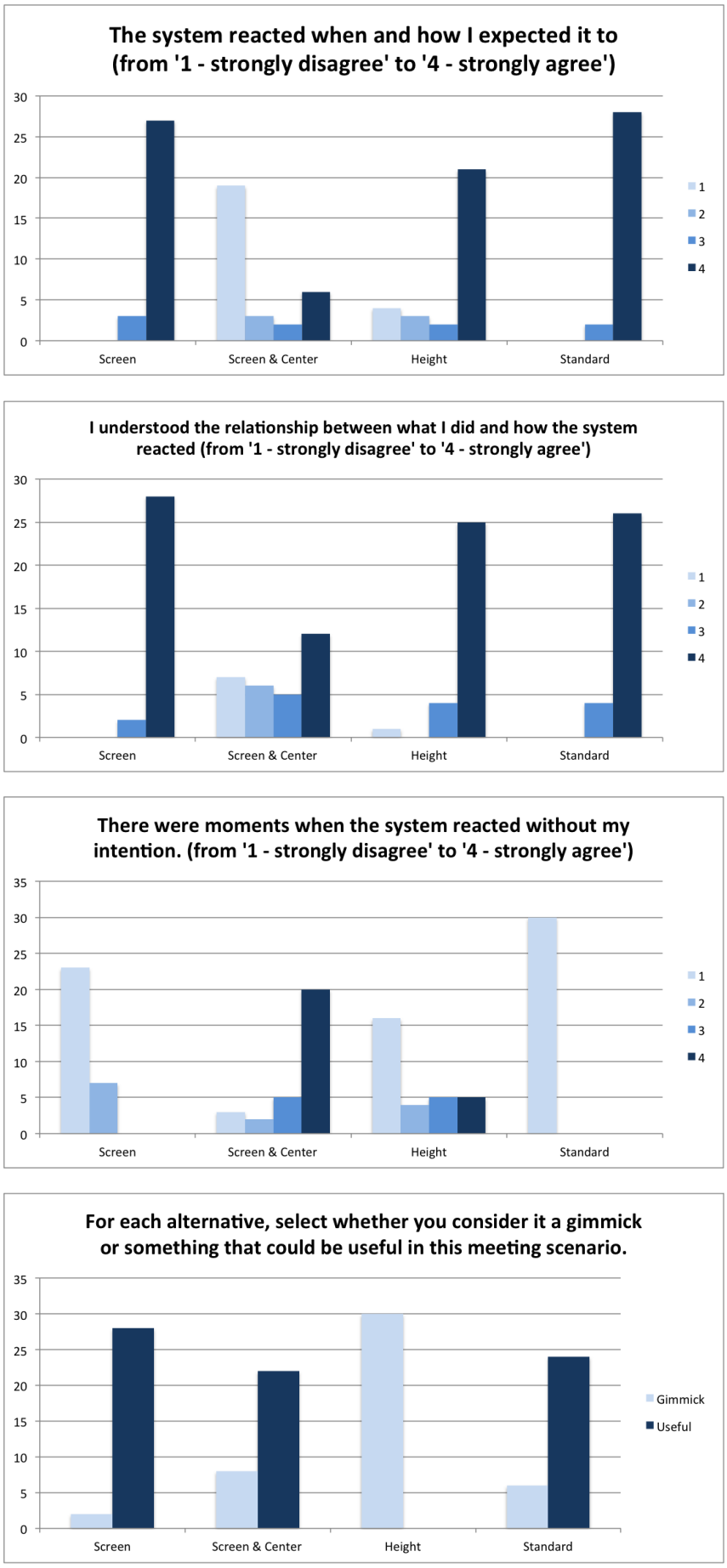


Figure F.2: Argument presentation questionnaire results (2/2).

Appendix G

Final Pre-Session Questionnaire

Pre-Session Questionnaire

* Required

Meeting Process Understanding

1. **Does it make sense to have an objective belonging to more than one dimension?** *
   Mark only one oval.
   - Yes
   - No

2. **Suppose that when you define a criterion for a particular objective you understand that the objective is not well defined. What should you do?** *

   _____

   _____

   _____

   _____

   _____

3. **Are the weight values relative or absolute?** *
   Mark only one oval.
   - Relative
   - Absolute

Software Understanding

4. **Can any user propose an objective?** *
   Mark only one oval.
   - Yes
   - No

5. **Can any user control the flow of the session (enter objective proposal mode, enter vote mode, etc.)?** *
   Mark only one oval.
   - Yes
   - No

Figure G.1: Final pre-session questionnaire, page 1.

6. **How do you control your "relevance" value?** *

7. **In what moments during the session is the shared screen affected by your "relevance" value?** *

Self-Assessment

8. **Assess how well you understood the meeting process** *
   Mark only one oval.
   - 1 - did not understand at all
   - 2
   - 3
   - 4 - understood completely

9. **Assess how well you understood how to use the software** *
   Mark only one oval.
   - 1 - did not understand at all
   - 2
   - 3
   - 4 - understood completely

Figure G.2: Final pre-session questionnaire, page 2.

Appendix H

Final Post-Session Questionnaire

Post-Session Questionnaire

* Required

Meeting Process Understanding

1. **Does it make sense to have an objective belonging to more than one dimension?** *
   Mark only one oval.
   - Yes
   - No

2. **Suppose that when you define a criterion for a particular objective you understand that the objective is not well defined. What should you do?** *

   _____

   _____

   _____

   _____

   _____

3. **Are the weight values relative or absolute?** *
   Mark only one oval.
   - Relative
   - Absolute

Software Understanding

4. **Can any user propose an objective?** *
   Mark only one oval.
   - Yes
   - No

5. **Can any user control the flow of the session (enter objective proposal mode, enter vote mode, etc.)?** *
   Mark only one oval.
   - Yes
   - No

Figure H.1: Final post-session questionnaire, page 1.

6. **How do you control your "relevance" value?** *

.....

.....

.....

.....

.....

7. **In what moments during the session is the shared screen affected by your "relevance" value?** *

.....

.....

.....

.....

.....

Self-Assessment

8. **Assess how well you understood the meeting process** *
Mark only one oval.

- 1 - did not understand at all
- 2
- 3
- 4 - understood completely

9. **Assess how well you understood how to use the software** *
Mark only one oval.

- 1 - did not understand at all
- 2
- 3
- 4 - understood completely

User Satisfaction

Figure H.2: Final post-session questionnaire, page 2.

10. **Regarding the ultimate goal of the session, how adequate do you consider the steps applied (defining objectives, defining criteria, defining metrics, assigning weights)?** *

Mark only one oval.

- 1 - Very inadequate
- 2
- 3
- 4 - Very adequate

11. **How satisfied are you with the results of the session?** *

Mark only one oval.

- 1 - Very dissatisfied
- 2
- 3
- 4 - Very satisfied

12. **How well do you consider that your contributions were taken into account by other users?** *

Mark only one oval.

- 1 - Very badly
- 2
- 3
- 4 - Very well

13. **How well do you consider each user was able to express his opinions?** *

Mark only one oval.

- 1 - Very badly
- 2
- 3
- 4 - Very well

14. **How interested were you in what was being discussed in the session?** *

Mark only one oval.

- 1 - Not interested at all
- 2
- 3
- 4 - Very interested

Figure H.3: Final post-session questionnaire, page 3.

15. **How well did you understand others' point of view?** *
Mark only one oval.

- 1 - Very badly
- 2
- 3
- 4 - Very well

16. **If you were the mediator at any point during the session, how easy was it to control the session?** *
Mark only one oval.

- 1 - Very difficult
- 2
- 3
- 4 - Very easy

17. **What is your opinion on the visual adaptation to "relevance" in the proposal screen?** *
Mark only one oval.

- 1 - Very bad
- 2
- 3
- 4 - Very good

18. **What is your opinion on the visual adaptation to "relevance" in the weight assignment screen?** *
Mark only one oval.

- 1 - Very bad
- 2
- 3
- 4 - Very good

Figure H.4: Final post-session questionnaire, page 4.