Designing for Multi-User Interaction in the Home Environment: Implementing Social Translucence

Karin Niemantsverdriet[1] **Mendel Broekhuijsen[1,2]** **Harm van Essen[1]** **Berry Eggen[1,2]**

[1]Eindhoven University of Technology
Department of Industrial Design
Eindhoven, The Netherlands

[2]University of Technology Sydney
Faculty of Design, Architecture and Building
Sydney, NSW, Australia

{k.niemantsverdriet; m.j.broekhuijsen; h.a.v.essen; j.h.eggen}@tue.nl

ABSTRACT
Interfaces of interactive systems for domestic use are usually designed for individual interactions, although these interactions influence multiple users. In order to prevent conflicts and unforeseen influences on others, we propose to leverage the human ability to take each other into consideration in the interaction. A promising approach for this is found in the social translucence framework, originally described by Erickson & Kellogg. In this paper, we investigate how to design multi-user interfaces for domestic interactive systems, through two design cases where we focus on the implementation of social translucence constructs (visibility, awareness, and accountability) in the interaction. We use the resulting designs to extract design considerations: interfaces should not prescribe behavior, need to offer sufficient interaction alternatives, and previous settings need to be retrievable. We also propose four steps that can be integrated in any design process to help designers in creating interfaces that support multi-user interaction through social translucence.

Author Keywords
Research-through-design; Interaction Design; Interactive Systems; Multi-user Interaction; Home Environment; Social Translucence.

ACM Classification Keywords
H.5.2 User Interfaces (User Centered Design; Theory and Methods); D.2.2 Design Tools and Techniques (User interfaces).

INTRODUCTION
Interactive systems for domestic use have become increasingly popular. Such systems are in dialogue with the user to provide services for everyday use, for example in entertainment, communications and home automation. Examples include media players to stream videos or photos from mobile devices to the television (e.g., Apple TV [40], Google Chromecast [17]), wireless speaker systems (e.g., Sonos [41], Bose Soundlink [42]), smart thermostats (Nest [43]), and connected lighting systems (e.g., Philip's Hue [44], Elgato Avea [10]). With Internet of Things [15] becoming more popular, this type of interactive systems are expected to appear even more in future homes. Because of their interactivity, these systems offer great opportunities in terms of flexibility, adaptability and personalization. We notice, however, that they often seem to be designed for individual use, while the domestic environment is much more social, and interaction with such systems often influences multiple users at once.

To prevent conflicts and unforeseen influences on others, interactive systems that are intended to be used by multiple users, should provide an interface that support users in taking each other into consideration when making interaction decisions. A promising approach towards mutual consideration in interaction is to implement the three constructs of the social translucence framework [8]: visibility, awareness and accountability. In this paper, we explore how social translucence can be used to design multi-user interfaces for interactive systems in the home environment. One of the interfaces resulting from the design cases can be seen in Figure 1.

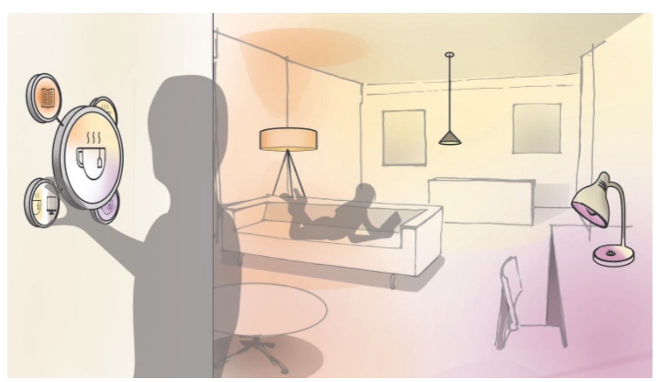

Figure 1. An impression of Orbit: a multi-user interface for interaction with networked lighting systems in the home environment.

Personal versus Multi-user Interaction

We see different characteristics in contemporary interactive systems for domestic use that make them less suited for multi-user interaction. First of all, interaction is often based on smartphone applications. The reasons for interaction designers to use smartphones are understandable: they offer great advantages in, e.g., computing power, mobility, and availability and their relative newness makes them appealing. However, smartphone interfaces make interaction highly personal [21,27]. Applications can be customized, and settings, playlists, and menu structures become incomparable between the different users of a single system, which could make settings of others irretrievable. Also, since smart phones and their connection to the system are not tied to a specific location, the person who is interacting can be rather invisible to other users. Furthermore, since often not everyone in the household is equally connected or involved in the interaction, system accessibility can be problematic (e.g., when children, grandparents, or nannies need control). Similar issues might occur when implicit interactions lead to, for example, lights turning off automatically when the main user is leaving the house even though other users might still be present.

Secondly, most interactive systems for domestic use offer much more complex control than their static predecessors. It seems likely to assume that when more control possibilities are presented, preferences between different users become more distinct, which could result in conflicts. When, for example, lights can only be switched on or off, there is little room for disagreement. But when options include color, saturation, brightness and spread of the light, preferences are likely to become further apart. Also, the routines around such systems become more flexible: there are no fixed settings needed anymore. Where previously a traditional light bulb had to be carefully selected to have the desired color temperature and luminance intensity, a contemporary connected light source can instantly switch between cool and warm light of any brightness. This makes it more difficult to establish agreements about the use of systems. So the advantage of flexibility those interactive systems offer, is at the same time likely to result in an increase of conflicts in use and preferences in multi-user situations.

Coordinating interaction between users

One often-named approach to resolve multi-user issues in interactive systems involves conflict-management [32]. Conflict management systems store information of use in user profiles and use this information to coordinate interaction possibilities for each user for pre-programmed contextual scenarios. When a conflicting preference is observed between different users (e.g., in deciding which channel to watch on TV), such systems can implicitly mediate between preferences and present an average alternative and/or request explicit user input (e.g., [18,34]). Another approach towards multi-user interaction makes use of hierarchies in control: a user that is higher in the hierarchy has more permissions than a person that is lower in hierarchy. These hierarchies are either determined by user profiles or by temporal or spatial priority (e.g., [28,33]). Hierarchical profiles result in coordination policies that prevent conflicts by limiting the interaction for some users over others [24] (e.g., presenting different interaction possibilities to parents and to children).

While these approaches are interesting from a technical perspective, they might not fit the social complexity of the home environment. Ethnographic research suggests that the home environment and the family life is highly dynamic and that routines are continuously adjusted [7,20]. This means that the scenarios of use, the user profiles, the preferences, and the hierarchies, on which all coordination strategies are based, become extremely complex and are unlikely to ever be complete. Also, permissions are not as static and many unforeseen exceptions are expected to arise: when a parent supervises the child, for example, the child might be perfectly allowed to perform certain interactions. Creating a system that can interpret and predict the context and successfully coordinate interactions for all users in all situations is virtually impossible [4].

Since an erroneous interpretation of the context is likely to lead to increased issues and conflicts, coordinating behavior in multi-user situations by a system might not be the best solution. We, therefore, propose to leave the coordination of interaction up to the users and to use system-mediation to support people in taking each other into consideration when deciding on an interaction.

Taking Each Other into Consideration

Estimating socially accepted behavior for different contexts might be very difficult for a computer program, but humans possess highly developed social skills to interpret social cues, opinions, behavior and intentions of other people. In daily life, people prevent and resolve arising conflicts through agreements, conversation, negotiation, and intervention all the time. Instead of snoozing, for example, people will quickly turn off the alarm so that they do not to wake others that are able to get up later. Or when someone is taking a shower, other people will wash their hands with cold water, if hot water cannot be taken from two taps at once. And when teenagers get home too late, they will walk on their toes and find their way in the dark, to not wake their parents. These familiar situations illustrate that people are very capable of coordinating actions in potential conflicting situations themselves.

Social Translucence

The process of coordinating actions to take each other into consideration in everyday life can be explained by the social translucence framework by Erickson and Kellogg [12]. This framework consists of three constructs: *visibility*, *awareness*, and *accountability*. *Visibility* explains how people use information from their surroundings to judge what appropriate behavior is. The provided information makes it possible for people to build up *awareness* of each other's actions, the intentions behind them, and the effect that interactions can have on others. In the example described above, when the lights are turned off and everyone has gone to bed, the person entering is aware that making noise will wake them up. Because of an understanding of action and reaction by all users, people can be held *accountable* for their actions. If the person entering is making noise even though s/he is aware that this noise will disturb others, the other people can ask for clarifications or might get angry. But when the others went to bed early unexpectedly and the person entering could not deduce this information from the context (because of a lack of visibility and awareness), s/he can also not be held accountable for accidently waking them. The social translucence framework describes that in order to support the users in coordinating actions amongst each other and to take each other into consideration, the system (or interface) should provide sufficient visibility, awareness and accountability. Important to note is that what information is visualized should be carefully considered because of the "vital tension between visibility and privacy" ([12] p.63): hence the term social *translucence*, instead of transparence.

The social translucence framework has mainly found resonance in the field of Computer Supported Collaborative Work (CSCW). Naturally so: the complete absence of visibility in distributed interaction in the digital world makes designed support indispensable. However, the nature of the digital realm has evolved since the introduction of the framework in the year 2000. With the type of interactive systems for domestic use that we discuss here, the physical and digital have merged into a complex hybrid. Therefore, we argue that a much broader adaptation of the framework would be in place, including not only distributed and work-related interaction but also interaction at home and possibly even multi-user interaction *in general*; regardless of the type of interface or the modality of interaction.

Aims & Contributions

In this paper, we investigate how to design multi-user interfaces for domestic interactive systems that leverage the human ability to take each other into consideration. We do so by implementing social translucence: interfaces should make socially salient information visible, leading to awareness and accountability in interaction. To investigate what considerations should be made in the interface and how such a perspective can be supported in the design process, we take a design approach. We present two design cases, where interfaces are designed for two popular domestic interactive systems: photo sharing and lighting. By comparing the resulting interfaces and by reflecting upon the design processes we aim to indicate questions, tools, and considerations that can help interaction designers implementing the social translucence framework.

In the remainder of this paper, we first review related work on social translucence. Next, we present the two design cases. We continue with reflecting on the cases and formulating design considerations for multi-user interfaces. We conclude this paper by reflecting on our design process, which leads to four steps that can help shift the focus of the design process towards implementing social translucence in the interaction.

RELATED WORK

Erickson's concept of social translucence originates from situations of geographically distributed communication in CSCW, which is why most of the related work can be found in this field. The main implementations of the framework are in the form of social proxies: visualizations that represent activity or participation in an interactive online context [1]. Babble [12] and Loops [11] are Erickson's social proxies that inspired the framework. They represent activity in different group communication scenarios, such as a chat room and an online lecture. Some of the many examples of social proxies from other researchers, include visualizations of user attention during conference calls [9,30,39] and visualizations of user participation in online auctions [29] (see the work of McDonald et al. [22] for an overview).

Insights from these research endeavors seem to have found their way in commercial interfaces: we see an increase of smartphone and web applications that have features to visualize the behavior of others. Think, for example, of the blue check marks in messaging application Whatsapp [38] that indicate that the recipient has read your message. Or the personally assigned cursor in Google docs that communicates where coworkers are editing text when working in a shared document [45]. Interesting about this last example, is that it integrates the visualization of information in the interaction itself (the cursor), while social proxies usually form a separate interface element.

While the majority of examples are found in geographically distributed scenarios, there are also examples of multi-user interfaces for collocated use. Here we also see the use of social proxies, especially in screen-based tabletop interaction. For example, the conversation Clock [5] displays individual contributions (measured from audio input) to a conversation on a tabletop projection, Messenger [8] presents speaker participation patterns in face-to-face discussions, and Narcissus visualizes activity levels which enables group members to evaluate individual contributions to collaborative work [36]. More interaction related, Reetz and Gutwin [31] investigated the visibility of collocated gestural interactions for personal mobile devices, and Tse et al. [35] designed gesture commands that provide consequential communication to other collocated people.

Important to note is that the large body of research on social devices (e.g., [6,19,21,27]) is not directly related to the work presented in this paper. Even though some of the resulting concepts form interesting examples of more multi-user focused interfaces, they pursue a different goal: social devices aim to trigger, enhance, and facilitate social interactions *between* people [19]. Instead, we want to support users in coordinating the interaction *itself*, regardless of the goal that the user has with this interaction. This goal might just as well be 'enjoying music', 'sharing photos', or 'turning on the light', as 'talking and engaging more with other people'.

To sum up: while there are a large number of examples of more socially translucent interfaces available, the framework has not been implemented broadly. Main examples include social proxies (visualizations of behavior) but a closer integration in the interaction itself might be more promising in some cases. Also, the implementations are to a large extent screen-based, although more tangible interaction styles could possibly portray the same qualities. In order to illustrate a broad implementation of social translucence in design, we focus one of our design cases on a *screen-based* smartphone application and one on designing a dedicated *tangible* interface.

DESIGN CASES

In order to come to an understanding of how to design for socially translucent multi-user interactions and to find guiding questions that support this focus in the design process, we design two new interfaces for well-known interactive systems for domestic use. Both design cases primarily aim at exploring the steps in the design process that help in designing for multi-user interaction, while creating example interfaces that demonstrate a broader implementation of the social translucence framework. The interface resulting from the first design case, *Shoto*, presents a streaming application for digital photo sharing in a social setting. The second design case results in *Orbit*, a tangible interface for a connected lighting system.

In both design cases we take a similar approach. We start with a brainstorm session with fellow design (researchers), during which we discuss the social translucence framework, come up with scenarios, and identify opportunities. The first two authors of this paper cluster and select the resulting ideas and iteratively develop the concepts further.

Case 1: Digital Photo Sharing in a Social Setting

With the coming of digital photos, sharing summer holiday photos with friends has been taken over by digital interfaces: instead of passing around our printed copies of photos, we now pass around our phones and tablets. In the early days of digital photography, Frohlich et al. [14] provided an overview of the requirements surrounding photo sharing technology, which they termed Photoware. They identified opportunities for sharing digital photos that relate to multiple users viewing and commenting on photos simultaneously. These types of challenges are still relevant to date, although new means of collocated photo sharing regularly appear in literature. Examples that allow multiple users to jointly interact with the system and influence the content that is shared, include: *4 Photos* [25], a concept for sharing Facebook photos at a dinner party; *Cueb* [16], a concept for sharing photos using two tangible cubes, to enhance communication between teenagers and parents; and *Shoebox* [2], a combination of photo storage device with a ambient photo display for the living room.

Despite these efforts in research, current commercial devices typically do not offer a dedicated interface for collocated photo sharing. However, there are technologies available to view photos on a larger screen, on smart televisions or popular domestic media streaming devices such as Apple TV [40] and Google Chromecast [17]. The photo is shared full-screen (either on the device or on a connected screen) by opening it in the photo management application of the (mobile) device. One of the scenarios of using systems like these is a presentation setting, with one dominant presenter and an audience of spectators. An alternative scenario (serving the people who are happy that the typical endless family gatherings in front of carousel slide projectors are getting rare) is a more dynamic photo sharing scenario. Multiple people can connect to modern media streaming device sequentially (when connected to the same network), allowing for users to continuously alternate between presenting and viewing/listening.

Multi-user Issues with photo streaming systems

In the shared scenario as described above, there are essentially four main *roles*: users could be talking about a photo (*narrator*), users could be searching for a photo and selecting it for sharing (*browser*), they could be sharing it on the screen (*sharer*), or they could be viewing and listening (*spectator*). While not all roles necessarily include interactions with the system itself (the spectator is not directly interacting); all these roles play a part in the multi-user setting and are influenced by interactions of others. Therefore, users need information about each other in order to decide on appropriate interactions.

In the current setup of media streaming devices, this multi-user interaction is hardly supported. Interactions of the different users have little visibility: when a photo is shared, it is at not obvious who the owner of the photo is. Also, there is no visibility of the intention behind the shared photo. Since every photo is shared full screen, there is no visible difference between photos that are intended to serve as an addition to the current narrative, as a comparison, or to introduce a new story, leading to low awareness. Of course, people will speak up to share this type of information but since the previous photo disappears when a new photo is shared there is no way to retrieve an image or to recover from an interaction that is undesired by other users. In addition, there is little accountability on, for example, attention: when someone is interacting with his/her mobile phone, s/he could just as well be typing a text message as searching for a new photo to share. So we can conclude that domestic media streaming devices currently do not present their users with sufficient information to coordinate behavior seamlessly amongst each other. In other words, the interaction is not socially translucent: there is not enough visibility of information to provide awareness and accountability.

Design Process

Our main goal in this design case is to design an interface that makes social photo sharing with media streaming devices more socially translucent. We aim at providing sufficient information in the interface to enable people to estimate appropriate behavior. The roles that we presented before (narrator, browser, sharer, and spectator) require different types of information to do so. A narrator would need awareness of the attention of the audience (are they interested in the story or are they distracted?) and about whether other people would like to share photos. A browser and a sharer would like to know when sharing (and interrupting the narrator) is appropriate. And the audience would like to be aware of the course of the story: who is sharing a photo and how does the photo relate to the story and the previous photos.

In order to translate this information into the new interface, we take inspiration from the more old-fashioned way of photo sharing: passing around printouts. In media streaming devices there is only one way to introduce a photo, namely full screen while removing the previous photo. With printouts, however, there are many more possibilities to share a photo. One could place his/her photo on top of the previous photo to take over the story or next to the previous photo to e.g. contrast the story. Also, in the act of placing the printout photo, intention can be expressed through the speed of movements, and the body posture of the person who is sharing. One could introduce the printout slowly, and hold it until the other person is finished talking, or slap it onto the table in one resolute movement. Both actions have very different consequences for the course of the story and the role of the current presenter and the interrupter. In the design of the interface we want to facilitate these different interaction opportunities by adding (1) different placements and (2) expression to the interaction.

Before a photo is shared, it needs to be selected from the personal collection. In photo sharing settings with printouts, this selection process is visible for the other people. Someone picks up his/her stack of photos, starts flipping through them, order the pictures, or take a desired photo from the stack. This photo might be placed immediately, but could also be placed in front of that person facedown to wait for the right moment. In this process, other people are aware that that person is looking for a photo to share in the near future. This means that it becomes possible to anticipate: if sharing of pictures is not desired at that moment, they can address that person to stop him/her from selecting new images. Important to note is that, while the activity of selecting is visible to others, the photos themselves are not. People can shield off the stack in the selection process and thus shield content of photos that are not being shared. This awareness of the intention to share photos is another element we want to bring into the new interface of media streaming devices.

We translate the relevant interactions from photo sharing with printouts to the interface of media streaming devices with the aim to make them more socially translucent. We do so by using the physical layout and actions of users with printouts as a metaphor. The resulting concept, called *Shoto*, is presented below.

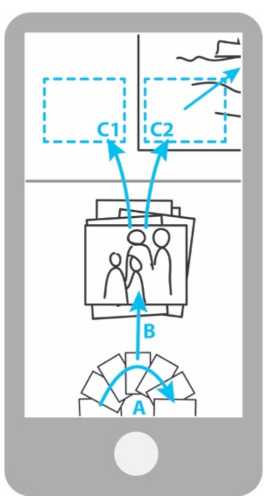

Figure 2. The personal interface of Shoto allows users to select a photo from their collection with the scroll wheel (A), drop it onto the stack to save it for later sharing (B), and drag a photo onto the canvas to share the photo. They can choose to place the image besides (C1) or on top of the previous photo (C2). The image can be scaled up by dragging out the corners.

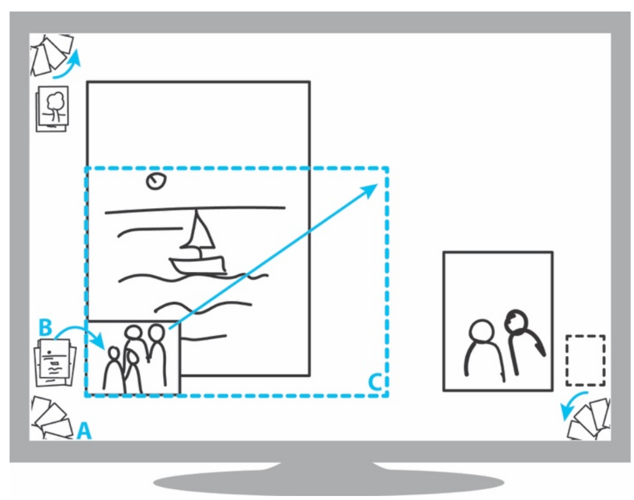

Figure 3. The shared interface of Shoto presents the scroll wheel (with blurred thumbnails; A) and stack with saved images for sharing (B) of each connected user. The central canvas shows (C) the collection of shared images in the size and position determined by the sharer. All interaction happens through the smartphones applications.

The Concept: Shoto

Shoto is an application for social photo-sharing sessions. The Shoto interface consists of two parts: a personal interface on each user's mobile phone where photos can be selected from the personal collection (Figure 2), and a collective interface for sharing on the central screen (Figure 3). The personal mobile phone interface consists of three parts: the scroll wheel (2A), a stack of selected photos to share (2B), and the canvas for sharing and presenting (2C). People can browse through their personal photo collection using the scroll wheel and drop images that are meant for sharing onto the stack. Photos from the stack can be shared on the screen by dragging them onto the canvas, either on top of the previous photo, or besides, after which the shared photo on the canvas can be resized. This is the translation of the different placements (covering the photo to change a story line, or placing next to it for comparison) and expression (in speed and final size when resizing the photo). All interaction happens through the smartphones applications. In the shared interface on the television screen, similar elements can be found: a scroll wheel (3A) and stack (3B) for every connected user, and the canvas where all shared photos are presented (3C). In this shared interface, the thumbnail images of the scroll wheel are blurred to make sure that content of unshared photos is shielded off. However, the activity of searching for photos is visible to the other users in this way. Similarly, the stack tells the other users that this person is planning to share photos soon and gives information about the size of the contribution.

Even though Shoto is conceptual, it presents a way of photo sharing that takes the multi-user situation into account. Some of its main advantages are that 1) users remain in control of the content that is shared; 2) there is diversity in ways of sharing for less intrusive additions; 3) since shared photos remain on the canvas interactions can always be undone. Next to being more socially translucent, Shoto might offer a more participatory and engaging way to share photos for all users involved.

Case 2: Networked Lighting Interaction

Networked LED light systems (e.g., Hue [44], WeMo [3], Avea [10]) are becoming popular within domestic environments. They offer much more flexibility in lighting use, since the wirelessly connected LED lights can individually be controlled on hue, saturation, and brightness. While implicit interaction with lighting is being explored (also in commercial applications, through phone sensor data), research has shown that user control in lighting systems is vital and results in much higher satisfaction [23,37]. Important in lighting control is to provide sufficient freedom while balancing effort [26]. A promising way to do so is through presets: pre-programmed light settings that address multiple lamps at once. Recently, some physical lighting preset controllers have started to appear on the market. The Hue Tap [44] presents three buttons for easy access to often-used presets. M-Qbe [46] is a cube where presets can be pre-programmed to its six sides. The cube can be rotated to adjust brightness. Main interaction with all above-mentioned networked lighting systems, however, relies on smartphone applications. In most applications, home screens also present preset-based interaction, with the possibility for users to adjust parameters of individual lamps.

Multi-user Issues with networked lighting systems

In lighting interaction, we see two main roles that users can have. If a person is directly interacting with the lighting system in order to change the light setting in the room, we call this person the *interactor*. Usually, there will be only one interactor at the time but there could be multiple people adjusting the light at once. All people present in the room, we call the light *consumers*: even though they are not interacting with the system directly, they are making use of the light, which makes them a user of the system. Of course, the interactor is also a consumer.

When looking at a family living room, where mother reads a book on the couch and her daughter is doing her homework at the kitchen table. Since presets usually influence all lamps simultaneously, when the mother adjusts the light to a 'reading' preset, she also changes the study light used by the daughter. In this situation with more traditional light switches, the mother would need to walk over to every light switch of every light she wants to control, and most likely, she would therefore only change the required light for her reading activity. Moreover, in walking towards the light switch of the dinner table, she would see that her daughter is using the table for studying (visibility & awareness) and she would probably discuss the light adjustments with her daughter since she would be accountable for making any undesired changes. From the daughter's perspective, it is much more visible that her mother changes the light (since she is near the switch) and she can therefore either intervene before the light is adjusted, or hold her mother accountable and ask her to undo any undesired adjustments. With the mobile phone interaction the interactor is much less visible, which means that it is not possible to intervene before the light is adjusted. And since users can create different personalized presets in their own application, it is often impossible to retrieve the previous light setting after a new preset is loaded. Lastly, it questionable that everyone has equal access to the lighting application (think of guests, young children), which introduces a whole new range of implications. To conclude, visibility, awareness and accountability in the smart phone interaction with digital lighting systems is very low.

Design Process

Our goal for this design case is to make an interface for networked lighting systems that make use of presets in a more socially translucent way. In order to increase accountability for the interactor with the light, we need to provide this user more information about the consumers.

The interactor would need to know whether consumers find it important to have the current light setting unchanged. Also, s/he needs to have a better understanding of the effect that an interaction will have on other people. In order to be able to act on that information, the interactor would need to have the opportunity for more detailed interaction: just one preset for the whole room makes it impossible to take others into consideration. For the consumers, we want to increase awareness that someone is actually planning to adjust the light. Finally, it is crucial to undo an undesired interaction, so retrieving the previous preset should be possible at all times.

The information requirements are translated into an interface for networked lighting systems for the living room. Since we aim to make a more tangible interface in this design case, many different possible translations are explored through quick mock-up prototypes, made with scrap materials. The variations are assessed on their interaction qualities and on the clarity of the translation. The resulting interface, called *Orbit*, is described below.

The Concept: Orbit

Orbit is a light interface for the living room, which offers both *global* control of the overall lighting preset that is applied in the living room, as well as *local* control of separate smaller areas of that room. The interface is located near the entrance of a living room. The interface consists of a *global display*, used to select global light presents, and smaller *local area displays* used to select lighting presets for specific areas in the room. In this way, users can manage the effect size of their interaction. The number of areas in which the living room should be divided can be adjusted per room, by adding extra area displays. In the example in Figure 1 (in the Introduction section) and Figure 5, we present an interface where four local area displays are attached: one for the lounge area with the couch, on for the open kitchen, one for the study area with a desk, and one for the children's play corner.

A selection ring surrounds each display. The ring can be turned to scroll through the different lighting presets and to preview them on the display (Figure 5A). Pressing the selection ring applies the selected preset to the corresponding area in the room. Because of this two-phased selection, users get an idea of the effect of the lighting adjustment on the room through the preview, before actually applying it in the room and influencing others.

By default, the area displays are physically connected to the global display. The area displays can be disconnected by pulling them away from the global display (5B). When activating a preset using the global display, all the connected area displays (and the corresponding lamps in those areas) are set to the light setting of the selected preset. The disconnected area displays are not influenced by global adjustments. They can be set individually to a certain preset through their selection rings, similar to the interaction with the global display (5C). By disconnecting an area display, a consumer can communicate the importance of specific interest in the lighting in that area.

Reconnecting an area by pushing it towards the global display is a conscious decision, which makes us believe that unintentional adjustments are less likely to be made. Therefore, people that do change the lighting preset in a disconnected area are accountable for their action. Similarly, when an undesired light change is made for the global setting, users are only entitled to hold others accountable when they have disconnected their area first: they cannot complain if they did not make visible that lighting changes in their area are undesirable. Furthermore, the preset icon that is loaded on the disconnected area display provides information about the reason for disconnecting. When an area is disconnected from the global display but not actively used in this specific way (e.g., the 'study' preset is loaded for the desk area, but the study books are gone from the desk), the area display can be reconnected again.

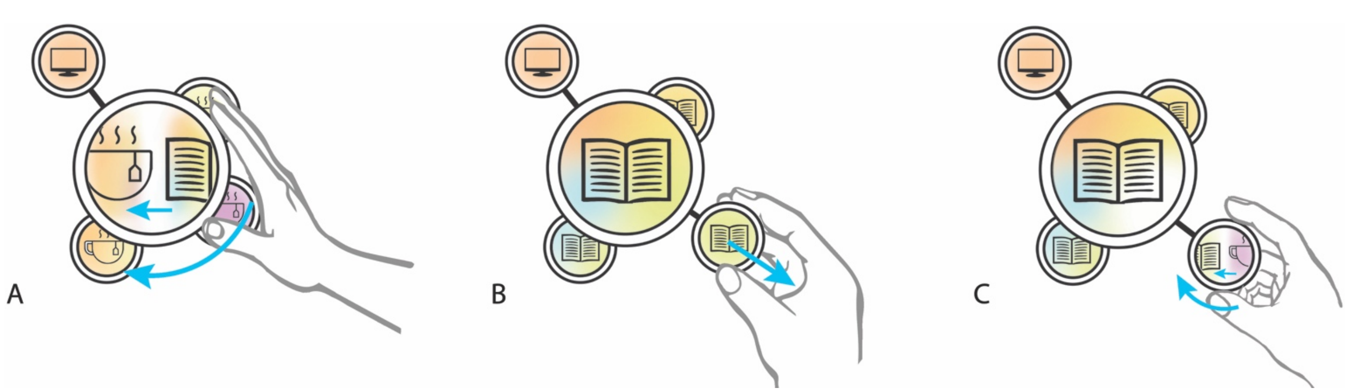

Figure 5. Interaction with Orbit. A. By turning the selection ring, users can scroll through and preview the different presets that are uploaded to Orbit. Pressing the ring activates the preset in all areas that are connected to the global interface. B. By disconnecting an area (by pulling it away from the global interface), it is not influenced by global lighting adjustments anymore. C. Disconnected areas can be separately set to a different preset, also by turning to select and pressing to activate.

Orbit is designed for everyday lighting interactions. For interactions that are not done on a day-to-day basis, such as linking lamps to the area displays or creating a new preset, a mobile phone application can be used. New presets need to be uploaded to the displays of Orbit before they can be used in the room, to ensure that everyone can access (and thus retrieve) the same presets.

DESIGN CONSIDERATIONS

The goal of the design cases is to better understand how to design for socially translucent multi-user interfaces. In this section, we present considerations for design that came up during development of Shoto and Orbit.

Systems Should not Judge Behavior

In Orbit and Shoto, we tried to design the interfaces in such a way that they do not prescribe or judge behavior. In line with Erickson's claim to portray actions, not interpretations [13], the concepts aim to leave the interpretation of what behavior is appropriate to the users themselves. In Shoto, this is translated into how people can, for example, engage in unrelated activities on their mobile phone at any time: their other applications are not disabled or shielded off. Orbit allows people to reconnect areas to the global display and to make adjustments to the lighting in every area, also in the ones that have been disconnected by other people. Reflecting on the interfaces, we see this as a key feature, since what is accepted varies: we could think of numerous examples where answering an urgent text message is perfectly acceptable although listening to a photo presentation. For systems, it would be virtually impossible to interpret the subtle differences in the situation only from contextual information (like whether a text message is more important than the current story that is being told). People do this type of estimations with ease and almost continuously. By increasing the social translucence of the interface, people can hold *each other* accountable for any undesired behavior, without the need for systems to interpret what is appropriate according to rigid rules .

Provide Sufficient Interaction Alternatives

In both design cases, we found that intricate interaction possibilities were required. Even if people have high awareness of each other's needs, if there are no interaction alternatives it becomes impossible to act according to this information. For example, when someone can only choose between changing all the lights and leaving the light unchanged, this person has little possibility to take others into account in the interaction. While accountability would still be possible, this situation will result in compromises: one of the users needs to settle for a different setting than s/he preferred. In Orbit, the interaction can be global and local, which opens up more detailed possibilities. So it seems that socially translucent interfaces should not only express information about other users but should also present sufficient interaction alternatives to mediate between the different users needs.

Allow Users to Return to Previous Settings

Another aspect we discovered in the design cases is the need to be able to return to a previous setting. The need to correct system's actions has been mentioned before (e.g., [4]) but it seems just as important to recover from other user's actions. Even with a more socially translucent interface, undesirable outcomes of an interaction are still possible: people can misinterpret the situation, interact without sufficient attention, or decide not to take others into consideration in their interaction, to name some examples. For accountability to work, returning to a more desired (previous) setting when an interaction is undesired should be possible. This means that in interactive multi-user systems, previous states of a system need to be retrievable. In Shoto, this is resolved by allowing the person who is about to share a photo to cancel his action with a simple gesture. In Orbit, all presets are uploaded to the shared interface, which means that all presets are available to, and thus retrievable by, all users.

Balance between Visibility and Privacy

To handle the issue of privacy, as mentioned by Erickson and Kellogg [12], we focused in the two design cases on the visibility of information about an *interaction*, instead of information about the *content*. In Shoto, for example, the scrolling wheel on the shared interface only presents blurred thumbnails of the photos. Only when a photo is placed on the stack, and thus intended to be shared, it becomes visible to others. Similarly, the canvas only indirectly shows if users are engaging in other activities: when a user is interacting with their mobile phone but there is no scrolling action visible, the other users can decide to ask for more information about the attention of that user. The canvas does not share this information about other activities that are performed with the mobile phone. In this way, concealing behavior with little white lies remain possible, which we see as a crucial part of social behavior. Furthermore, we made sure that all information is always visible to *all* users, so each user also knows what information is shared about him/her. This consideration is in line with Erickson [13] who states that every user in the system should have the same information. We think that responsibility for guarding privacy in socially translucent multi-user interfaces lies with the designers, who should carefully evaluate the trade off between the need for information and the privacy of users.

DESIGNING FOR MULTI-USER INTERACTION

In the previous section we described considerations resulting from the designs of Orbit and Shoto. When we look at the design processes of both cases, we can distinguish commonalities in the questions that we considered. We identify four steps that we found useful for implementing the social translucence framework (see Table 1). Below we describe each of the steps from Table 1 in more detail.

1. Describe User Roles within the System

As discussed in the cases, people can have different roles in the interaction in different use cases. In Shoto, for example, there are three roles in a typical moment in time: the 'sharer', the 'browser' and the 'passive spectator'. Usually, people switch between roles, and multiple people can have the same or different roles at the time. So roles are not exactly the same as stakeholders or users. A user interacts with a system, where a role can be a person who is involved in the system without interacting (e.g., the role of 'consumer' in Orbit). It is important to have a complete overview of *all* roles that are part of the envisioned system.

2. Determine the Need for Multi-User Interaction

Once all roles are identified, it becomes possible to see whether roles influence each other. This determines whether there is a need for a multi-user focus in the design of the interactions. In Shoto, the multi-user situation is quite clear: the different roles influence each other through simultaneous interaction: the canvas that the presenter is discussing changes when the sharer introduces a new photo. In Orbit, there is usually only one person interacting with the interface to change the light. Light consumers, however, are influenced by the light change, even though they are not actively using the system.

If there is no influence of an interaction on any of the roles, there is no need for visibility, awareness, or accountability for that interaction. Even in multi-user situations some interactions do not have direct influence on others (e.g. browsing through the preset in Orbit), which means that such interactions do not require information sharing to other users. So in this step, the relations between the roles and the situations in which roles can influence each other need to be identified, in order to determine the need for a multi-user focus.

3. Define Required Information

Once the interactions that influence other users are determined, the information requirements can be determined. The person that is interacting needs to be able to estimate what appropriate behavior is for him/her to be accountable. This estimation can only be done if s/he is aware of the intentions of others. For example, in Orbit, the user changing the lighting conditions needs to know whether the people using the light have a desire to maintain the current lighting condition. After the lighting conditions have changed, people using the light need to know who adjusted the light to address him/her if this was undesired. It is important to be specific about what information is required, since, as discussed before, providing too much information could lead to privacy issues. As a tool to ask the right questions, Bellotti and Edwards' list on "human-salient details" for context-aware systems could be of use. These details consist of presence, identity, arrival, departure, status, availability, and activity [4].

4. Translate Information into Interaction Aspects

Lastly, the specified information needs to be translated into the interaction. While the social translucence framework speaks of *visibility*, information does not necessarily need to communicated visually: the information could also be translated visually, haptic, or auditory, for example; depending on what suits a context and concept. So design skills are important in this translation process. In Shoto, we looked at interaction possibilities with printout photos to inspire the translation of information in the interface. We found that in this analogue interaction all the information requirements were covered, and translating the analogue interaction to a digital one helped us to make the interaction with Shoto understandable and complete. In Orbit, we took inspiration from the traditional situation of switching

Process Step	Example Questions to be Asked by the Designer
1. Understand which users are part of the system, by describing the different user roles	*Who are the users of the system?* *How many different kind of users (roles) can interact with the system?* *Are these roles using the system at the same time, or one at the time?* *Can a user have more than one role, or are they mutually exclusive?*
2. Determine the multi-user interaction need, by mapping the influence of every interaction on all the roles	*For every interaction, is there an influence on other users?* *If so, how are the other users influenced?*
3. Define the information different users require to take each other into consideration in the interaction	*Which information is needed for every role to create awareness of other users?* *Which information is needed for every role to be held accountable for actions?* *What information is currently available (visible) for the different users?*
4. Translate the required information that lead to enhanced social translucence into interaction aspects	*How can the required information be translated into interaction aspects?* *Which social interactions can inspire the translation of required information?* *Should the information be available centrally, locally, or both?* *Considering user privacy, is all information necessary in its current form?*

Table 1: steps to include into any iterative design process to design for multi-user interaction. In the table the left column describes the process step, the right column lists example questions that can be asked to aid the design process

individual lamps on or off, while having a centrally placed switch near the entrance. With these two cases we found that looking at alternative interaction examples from similar or other domains can be useful in the translation process.

DISCUSSION

Orbit and Shoto are conceptual and have not been empirically evaluated, so we can only imagine how the interaction will actually be used in the real context. Other important criteria, such as feasibility and cost-effectiveness were not taken into account. Also, we cannot conclude if the resulting interactions are an improvement compared to existing interactions or not. Instead, we were able to draw from the process of designing them, to highlight topics that we believe require consideration in the design of interfaces for multi-user interaction.

The four steps that we propose are not necessarily new to the design practice: defining stakeholders, scenarios of use, and interaction requirements are integral parts of any design process. They appear to be typical for any design process, but because the steps and questions are tailored towards specifically addressing social translucence aspects in the interaction, we believe they can help interaction designers to shift the general focus of the design activity to include multi-user interaction. Our aim is change the designer's attitude towards multi-user interaction, and because these steps can be used alongside their own design process, they should be treated as an addition instead of a replacement.

Moreover, it is also important to note is that following these steps does not automatically result in a successful interface, also because the example questions that we pose do not lead to straight answers. Although it can help designers to shift their focus towards social translucence, we would like to emphasize that a solid understanding of the social translucence constructs is required.

The need for implementing a better support for multi-user interaction is also relevant in other shared environments than the home, such as offices, hospitals, and public spaces. Although the work is this paper has been focusing on interaction design in the domestic environment, we feel that the four steps we propose can be used for more general multi-user interaction design challenges.

CONCLUSIONS

The work presented in this paper centers around the observation that interactive systems for domestic use are usually designed for individual interactions, whereas in many situations these interactions involve multiple users. We believe that, with interactive systems becoming more and more present in our daily life, it is essential to design interfaces that leverage the human ability to take each other into consideration. To support designers in creating systems for multi-user situations, we propose a broader implementation of the social translucence framework, as originally described by Erickson & Kellogg, to include not only distributed and screen-based situations but all situations of multi-user interaction with interactive systems for the home environment.

To investigate *how* to design for such more socially translucent interfaces, we performed two design cases, where we made socially salient information visible in the interfaces to increase awareness and accountability in the interaction. By reflecting upon the cases, we have derived a number of considerations for multi-user interfaces and four steps to support designers that aim to integrate better support for multi-user situations in their designs. We found that interfaces should not judge or prescribe behavior, that they need to offer sufficient interaction alternatives, and that previous settings need to be retrievable in order to recover from undesired results of interactions. To support the design process, we found that (1) defining roles, (2) for each role describing influence of interaction on other roles, (3) defining information requirements, and (4) translating this information into the interface are steps that can be integrated in any design process to ensure a multi-user focus. The steps facilitate designers in asking the right questions and to iteratively pay attention to multi-user implications of their choices. Design skills play an important role in the translation process, and designers need to carefully balance between privacy and information needs.

With our work we contribute to shift the focus that we believe is needed in the design of multi-user interaction. The steps that we present can help interaction designers to integrate this focus within the design process, and we hope to inspire future work on design for multi-user interaction.

ACKNOWLEDGEMENTS

We thank the participants in the brainstorm sessions for their inspiring input in the case studies: Pengcheng An, Nikolaos Batalas, Hanneke Hooft van Huysduynen, Meis Suker, Qi Wang, and Jiang Wu. Part of this research was supported by STW VIDI Grant No. 016.128.303 of The Netherlands Organization for Scientific Research (NWO), awarded to Elise van den Hoven.

REFERENCES

1. Mark S. Ackerman and Brian Starr. 1995. Social activity indicators. *Proceedings of the 8th annual ACM symposium on User interface and software technology - UIST '95*, ACM Press, 159–168. http://doi.org/10.1145/215585.215969

2. Belkin. WeMo Home Automation. Retrieved August 4, 2015 from http://www.belkin.com/us/Products/home-automation/c/wemo-home-automation/

3. Victoria Bellotti and Keith Edwards. 2001. Intelligibility and Accountability: Human Considerations in Context-Aware Systems. *Human-Computer Interaction* 16:2-4, 193–212. http://doi.org/10.1207/S15327051HCI16234

4. Tony Bergstrom and Karrie Karahalios. 2007.

Conversation Clock: Visualizing audio patterns in co-located groups. *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)*, IEEE, 78–78. http://doi.org/10.1109/HICSS.2007.151

5. James Clawson, Amy Voida, Nirmal Patel, and Kent Lyons. 2008. Mobiphos. *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services - MobileHCI '08*, ACM Press, 187. http://doi.org/10.1145/1409240.1409261

6. Scott Davidoff, Min Kyung Lee, Charles Yiu, John Zimmerman, and Anind K. Dey. 2006. Principles of Smart Home Control. In *UbiComp 2006: Ubiquitous Computing*, Paul Dourish and Adrian Friday (eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 19–34. http://doi.org/10.1007/11853565

7. Joan Morris DiMicco, Katherine J. Hollenbach, Anna Pandolfo, and Walter Bender. 2007. The Impact of Increased Awareness While Face-to-Face. *Human–Computer Interaction*. Retrieved June 24, 2015 from http://www.tandfonline.com/doi/abs/10.1080/07370020701307781#.VYpfqGDSK9s

8. Xianghua Ding, Thomas Erickson, Wendy A. Kellogg, et al. 2007. An empirical study of the use of visually enhanced voip audio conferencing. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*, ACM Press, 1019. http://doi.org/10.1145/1240624.1240780

9. Elgato. Avea. Retrieved August 4, 2015 from https://www.elgato.com/en/smart/avea

10. Thomas Erickson, Wendy a Kellogg, Mark Laff, et al. 2006. A persistent chat space for work groups: the design, evaluation and deployment of loops. *Proceedings of DIS'06: Designing Interactive Systems: Processes, Practices, Methods, & Techniques*, 331–340. http://doi.org/10.1145/1142405.1142454

11. Thomas Erickson and Wendy A. Kellogg. 2000. Social Translucence: An Approach to Designing Systems That Support Social Processes. *ACM Trans. Comput.-Hum. Interact.* 7, 1, 59–83. http://doi.org/10.1145/344949.345004

12. Thomas Erickson. 2003. Designing visualizations of social activity: six claims. *CHI '03 extended abstracts on Human factors in computing systems - CHI '03*, ACM Press, 846. http://doi.org/10.1145/765891.766027

13. Daniel Giusto, Antonio Iera, Giacomo Morabito, and Luigi Atzori (eds.). 2010. *The Internet of Things*. Springer-Verlag New York. http://doi.org/10.1007/978-1-4419-1674-7

14. Google. Chromecast. Retrieved June 24, 2015 from https://www.google.com/intl/nl_nl/chrome/devices/chromecast/index.html?utm_source=chromecast.com

15. Md. Kamrul Hasan, Kim Anh Pham Ngoc, Young-Koo Lee, and Sungyoung Lee. 2009. Preference learning on an OSGi based home gateway. *IEEE Transactions on Consumer Electronics* 55, 3, 1322–1329. http://doi.org/10.1109/TCE.2009.5277995

16. Pradthana Jarusriboonchai, Thomas Olsson, and Kaisa Väänänen-Vainio-Mattila. 2013. Roles, scenarios and challenges of social devices. *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication - UbiComp '13 Adjunct*, ACM Press, 1575–1578. http://doi.org/10.1145/2494091.2497364

17. Min Lee, Scott Davidoff, John Zimmerman, and Anind Dey. 2006. Smart Homes, Families, and Control. *Research Showcase @CMU*. Retrieved June 15, 2015 from http://repository.cmu.edu/hcii/226

18. Andrés Lucero, James Clawson, Kent Lyons, Joel E. Fischer, Daniel Ashbrook, and Simon Robinson. 2015. Mobile Collocated Interactions. *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '15*, ACM Press, 2437–2440. http://doi.org/10.1145/2702613.2702649

19. David W. McDonald, Stephanie Gokhman, and Mark Zachry. 2012. Building for social translucence. *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12*, ACM Press, 637. http://doi.org/10.1145/2145204.2145301

20. Bernt Meerbeek, Marije te Kulve, Tommaso Gritti, Mariëlle Aarts, Evert van Loenen, and Emile Aarts. 2014. Building automation and perceived control: A field study on motorized exterior blinds in Dutch offices. *Building and Environment* 79, 66–77. http://doi.org/10.1016/j.buildenv.2014.04.023

21. Meredith Ringel Morris, Kathy Ryall, Chia Shen, Clifton Forlines, and Frederic Vernier. 2004. Beyond "Social Protocols": Multi-user Coordination Policies for Co-located Groupware. ACM, 262–265. http://doi.org/10.1145/1031607.1031648

22. S.A.M. Offermans, H.A. van Essen, and J.H. Eggen. 2014. User Interaction with Everyday Lighting Systems. *Personal and Ubiquitous Computing*, 1–21. http://doi.org/10.1007/s00779-014-0759-2

23. Thomas Olsson, Pradthana Jarusriboonchai, and Kaisa Väänänen-Vainio-Mattila. 2015. Towards Wearable Displays Aiming to Enhance Social Interaction. *Submission to CHI'15 workshop Mobile Collocated Interactions: From Smartphones to Wearables*. Retrieved June 11, 2015 from

http://www.funkydesignspaces.com/mobile_collocated/wp-content/uploads/2015/02/03-Olsson.pdf

24. Filip Petrushevski. 2012. Personalized lighting control based on a space model. *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12*, ACM Press, 568. http://doi.org/10.1145/2370216.2370311
25. Sheizaf Rafaeli and Avi Noy. 2007. Social Presence: Influence on Bidders in Internet Auctions. *Electronic Markets*. Retrieved June 24, 2015 from http://www.tandfonline.com/doi/abs/10.1080/10196780500083886
26. Rahul Rajan, Cliff Chen, and Ted Selker. 2012. Considerate supervisor. *Proceedings of the 2012 ACM annual conference extended abstracts on Human Factors in Computing Systems Extended Abstracts - CHI EA '12*, ACM Press, 2609. http://doi.org/10.1145/2212776.2223844
27. Adrian Reetz and Carl Gutwin. 2014. Making big gestures. *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, ACM Press, 4087–4096. http://doi.org/10.1145/2556288.2557219
28. Choonsung Shin, Anind K. Dey, and Woontack Woo. 2010. Towards combining automatic resolution with social mediation for resolving multiuser conflicts. *Cybernetics and Systems* 41, 2, 146–166. http://doi.org/10.1080/01969720903584282
29. Choonsung Shin and Woontack Woo. 2005. Conflict Resolution based on User Preference and Service Profile. *IPSJ SIG Technical Reports* 2005, 60, 311–316.
30. Edward Tse, Chia Shen, Saul Greenberg, and Clifton Forlines. 2007. How pairs interact over a multimodal digital table. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*, ACM Press, 215. http://doi.org/10.1145/1240624.1240659
31. Kimberley Upton and Judy Kay. 2009. Narcissus: Group and Individual Models to Support Small Group Work. In *User Modeling, Adaptation, and Personalization*, Geert-Jan Houben, Gord McCalla, Fabio Pianesi and Massimo Zancanaro (eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 54–65. http://doi.org/10.1007/978-3-642-02247-0
32. Jennifer a. Veitch. 2001. Psychological processes influencing lighting quality. *Journal of the Illuminating Engineering Society* 4480, 1, 124–140. http://doi.org/10.1080/00994480.2001.10748341
33. WhatsApp. WhatsApp FAQ - How can I tell if someone has read my message? Retrieved June 18, 2015 from https://www.whatsapp.com/faq/android/28000015
34. Nicole Yankelovich, William Walker, Patricia Roberts, Mike Wessler, Jonathan Kaplan, and Joe Provino. 2004. Meeting central. *Proceedings of the 2004 ACM conference on Computer supported cooperative work - CSCW '04*, ACM Press, 419. http://doi.org/10.1145/1031607.1031678
35. Apple - Apple TV - HD iTunes content and more on your TV. Retrieved June 24, 2015 from https://www.apple.com/appletv/
36. Sonos | HiFi Wireless Speakers & Home Audio Systems. Retrieved June 24, 2015 from http://www.sonos.com/
37. Wireless Speakers. Retrieved September 22, 2015 from https://www.bose.com/products/speakers/wireless_speakers.html
38. Nest. Retrieved September 22, 2015 from https://nest.com/
39. Hue Tap. Retrieved June 24, 2015 from http://www2.meethue.com/en-us/the-range/hue-tap/
40. Google Docs - create and edit documents online, for free. Retrieved June 18, 2015 from https://www.google.nl/docs/about/
41. m-Qbe. Retrieved November 24, 2015 from http://m-q.be/