THE NEEDFINDING MACHINE

A DISSERTATION
SUBMITTED TO THE DEPARTMENT
OF MECHANICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Nikolas Martelaro
February 2019

Abstract

This thesis develops and evaluates a method for allowing designers to perform needfinding by remotely interacting with users in real-time *through* computer-enabled interactive devices. While interactive systems present new opportunities for creating intelligent products that adapt to their users, these devices may also help designers understand people's needs by providing a view into users' everyday experiences. This thesis asks: how can designers use interactive devices to connect with users to receive information and feedback about their in-context experiences with interactive products? To answer this question, I develop the *Needfinding Machine*. Using the Needfinding Machine method, a designer in-the-loop observes the user, asks questions through an interactive machine, and remotely performs the machine's behaviors in real-time as a way to conduct problem-space exploration in-context.

This thesis has four major components. Part one gives an overview of the Needfinding Machine and situates the method among other design methods for understanding people and their needs. Part two presents a lab-based study exploring how we can use machines that speak to elicit meaningful information from people. I then move out of the lab and evaluate how practicing designers can use a needfinding machine to understand their users. Part three develops a needfinding machine for use in automobiles called WoZ Way. I then discuss a case study with designers from Renault using WoZ Way to understand driver experience with currently available advanced driving assistance features. Finally, part four presents a case study with an interaction research team from Spotify using a needfinding machine I developed, called DJ Bot, to explore people's experience with music in two contexts, the car and the home. These projects present a proof-of-concept for using interactive devices for real-time remote needfinding and describe how designers can engage with the Needfinding Machine method.

Acknowledgments

This thesis is the culmination of thinking about how we can use new interactive technologies to augment our abilities to do design work. As I have gone through my Ph.D., I have often found myself making connections to how my mentors and friends have influenced my thinking. There are many people I would like to acknowledge who have helped me during my journey toward this dissertation.

Years ago, I was introduced to engineering design by my high school teacher and mentor Mr. Joseph Carpenter. It was in his courses that I developed my skills and fondness for engineering design. Throughout my years in high school, Mr. Carpenter helped me build an aptitude for thinking creatively about different engineering challenges. In addition, it was Mr. Carpenter who taught me the value of rapid prototyping. I remember working through many iterations of ball grippers and vision systems for our small Mars Rover LEGO bots. Since working with Mr. Carpenter, I have always aimed to quickly see if an idea *could* work. These lessons about rapid prototyping and testing have served me well throughout my work and formed the bedrock for the work in this thesis. Aside from teaching me about engineering design, Mr. Carpenter also helped me decide to attend the Franklin W. Olin College of Engineering. This ultimately set me on the path to where I am today.

Olin was quite a unique place to go to undergrad. Specifically, the significant focus on design was something that thoroughly shaped my interests. I remember in my first-year Design Nature course, how I realized that engineering design was exactly the type of work I wanted to be doing. Moreover, I was driven to find out more about how we could improve our abilities to design and immediately sought out to start researching with one of our studio professors. This drive led me to a fruitful partnership with Dr. Özgür Eris studying how designers could use remotely connected whiteboards to collaborate with each other.

During the three years I worked with Özgür, I began to start thinking about how we work as designers and what tools might help us during our design process. Özgür helped me to start looking at what designers were doing and specifically how they were interacting with one another. He also taught me how conversations were much more than just spoken words and that they also included gesture and interaction with objects. Özgür helped me think about a new way to see design interactions and taught me how to capture these interactions as data that could be reviewed later. This way of thinking and the associated skills of building out video capture systems was hugely influential on how I developed the Needfinding Machine, laying the foundation for all the video and data capture systems that I have built since.

In addition to my work with Özgür, I had many other faculty members who greatly influenced how I think about design. Specifically, Dr. Ben Linder taught me that design could be a semi-structured process and that we could learn to be better designers. The courses I took with Ben greatly expanded the types of people whom I considered designing for. His courses taught me how powerful and challenging real needfinding can be. Moreover, Ben taught me what it meant to focus deeply on designing for needs. I often find myself still learning from our experiences in Alabama and Ghana while working on affordable design projects.

The other design professor I would like to thank is Lawrence Neeley. Lawrence taught me the value of saying "*yes, and...*" while designing. At the same time, his critique often made me think about things in completely new ways and fostered my creativity. Lawrence also championed creating physical artifacts as a way of building understanding. This type of thinking has stuck with me and I always feel that working with a material and doing something is a way to build understanding when trying to understand a design space or a research area.

The other professor at Olin who significantly influenced my research path was Dr. Jonathan Adler. His course on narrative identity introduced me to personal stories as way to understand people. I have always been one to tell and enjoy stories. However, Jon was the one who made me realize that the stories people tell about themselves could be used methodically to understand their personal motivations. The use of personal storytelling is core to how the Needfinding Machine functions as a way to provided designers with rich material to understand who a person is and how that may define their interaction with new technology.

My experiences at Olin also guided me toward pursuing my Ph.D. It was Özgür who suggested that I go to the Center for Design Research to study with Larry (both Özgür and Lawrence were Larry's students). I remember when I first arrived at Stanford. The then acting director of the DesignX group, Martin Steinert, welcomed me with open arms. Knowing my connection with Özgür and Lawrence, his first words to me were "Welcome home!" And for me, it did feel like coming home. Larry's group within the Center for Design Research was exactly the place I wanted to be for my Ph.D. I cannot express how impressed I am with the entire ecosystem that Larry has created for allowing researchers to explore ideas around what designers do when they design. I am very fortunate for the freedom I had working with Larry. I feel that his ability to allow students to wander while always being available was very conducive to my research explorations.

When I started on my Ph.D. journey, I had the great fortune to work with Dr. Malte Jung and Dr. Clifford Nass on a project that became the TutorBot studies. This project got me interested in Human-Robot Interaction. The TutorBot studies were also the catalyst for my thesis explorations as it was during the first set of TutorBot studies that I realized that we could use robots to elicit stories from people that could be useful for designers. My work with Malte also led me to collaborate with Dr. Pamela Hinds, who has since become one of my most trusted advisors with her steady and practical guidance on the various aspects of academic life.

Over the years, I have had many wonderful undergraduate and masters students who have helped me with my work. While doing the first set of TutorBot studies, I was also fortunate to work with Halsey Hoster. She was an amazing research assistant and helped me to run participants. Additionally, Halsey is the student who has helped me most to learn what it takes to advise researchers. I cannot thank her enough for the help she provided me during my first few years at Stanford. During my second year, Heather Altman was an essential member of a research team that helped me run an HRI study that was not a part of the thesis, but that was formative in my work within HRI. While running the second TutorBot study, I worked with the wonderful Victoria Nneji, who was critical in running participants and providing a thoughtful eye on how the students interacted with the robot. Her structured approach to working with me proved very helpful for the smooth operation of our study and of the final publication that resulted. I am also grateful to Catherine Smith

for her help with analyzing and coding how meaningful the student responses to the robot were. Much of the technical subsystem for the Needfinding Machine built upon the work of Victor Chahuneau when he developed the first iteration of our SofaBot remote control code. I fondly remember spending time with Victor and our friend and collaborator Marco Spadafora during the summer of 2015. Our work on remote controlled furniture ultimately led me to develop remote-controlled interfaces for real-world interactive devices. There are also others who have helped me to run studies both present and not present in this thesis. I am thankful for the help I received from Daniel Greer, Quillan Smith, and Jade Fernandez.

I am indebted to the many practicing designers who either participated directly in deployments of the Needfinding Machine or who provided feedback on the method and its relevance to design. These people include: Nikhil Gowda, Pierre Delaigue, Lionel Cordesses, Carrie Bobier-Tiu, and Holly Russel from Renault Silicon Valley; Melissa Cefkin and Erik Vinkhuyzen from Nissan Research Center Silicon Valley; Henriette Cramer, Jennifer Thom, and Sarah Mennicken from Spotify; and Carey Smith, Saran Borude, Esther Jun Kim, Maddie Douglas and Crystal Tobias from the Volkswagen Group of America.

My time at the Center for Design Research was enriched by many of my fellow labmates and friends. Whether in the lab, in class, or outside, the many conversations we had shaped my thinking and helped me grow as a researcher. Members of the DesignX lab include Sonia Baltodano, Hamish Tennent, Kyu Keogh, Andrea Cuadra, Mishel Johns, Srinath Sibi, Dave Miller, Dylan Moore, Pablo Paredes, David Sirkin, Becky Currano, David Goedicke, Ade Mabogunje, Neeraj Sonalkar, Burton Lee, and Phillip Preyer. My other friends from Stanford include Scott Henkelman, Megan Lin, Carolyn Estrada, Adi Rao, Jo-Ann Desais, Stephanie Tommasetta, Arul Suresh, and Buddy Gardineer. Finally, there have been other members of our design research community whom I admire and have provided me with guidance on this thesis. I would like to thank Bill Verplank, Hugh Dubberly, and Paul Pangaro for taking time on various occasions to speak with me about my work. I would like to thank James Landay for providing me with directed and actionable feedback on the contributions and implications of my research. Additionally, I would like to thank James for being on my reading committee.

I would like to thank my mother and father, Sheila and Sam Martelaro, for fostering in me the work ethic and determination that is required to complete this level of work. Their example has always inspired me to continue working for something bigger than myself. I also want to thank my mother and father for having the trust in me to define my own path over the years.

Finally, I would like to give my sincerest thanks to Dr. Wendy Ju for her guidance throughout my process of defining and developing the Needfinding Machine. It was Wendy who first clued me into the idea of "interaction design" and led me to the area that I feel I was on the periphery of for many years. My conversations with her have fundamentally shaped the way I think about design and design research. Working with Wendy reminds me of Rule 5 from *The Immaculate Heart College Art Department Rules* by Corita Kent: *"Be Self-Disciplined. This means finding someone wise or smart and choosing to follow them. To be disciplined is to follow in a good way. To be self-disciplined is to follow in a better way."* During my years at Stanford, I feel that I have been able to follow along with Wendy and now have reached a point where we are true collaborators. While this may be the end of one set of work, I look forward to many more years of collaboration.

Glossary

Cybernetics - The science of communication and control in the animal and the machine. Pertains to systems which have goals and use feedback to reach these goals.

HRI - Human-robot interaction

HCI - Human-computer interaction

Interaction - Reciprocal action between one or more entities. In this thesis, interaction focuses on people's interacting with other objects and environments.

Interaction Engine - My hardware and software system incorporating single-board computers, microcontrollers, and open-source software which can turn regular objects into interactive objects.

Interactive Systems - Designed objects or systems that can interact with people and other objects. These systems typically include some computational elements that enable them to take in information and take action in the world.

Internet of Things (IoT) - Products which are connected to the internet and typically have some data capture and processing abilities which determine how they interact with the world.

Needfinding - A structured design activity whereby designers attempt to understand the unmet needs of their users to help them generate product ideas.

Needfinding Machine - A method for using interactive devices to understand user needs in relation to a specific context. A needfinding machine is embedded in some product or device that itself is embedded in the user's environment and in their everyday life.

Ubiquitous Computing (Ubicomp) - Computing that is embedded in many types of everyday products.

Wizard of Oz - A prototyping method employing people hidden behind the scenes controlling new technology.

Contents

List of Tables

List of Figures

Chapter 1

Needfinding for the 21st Century

Imagine that you are a designer tasked with creating a range of "smart" kitchen appliances. These appliances will be interactive, intelligent, and provide capabilities beyond the basic function of the product. Maybe a coffee machine that tells you your morning news or a fridge that keeps track of your groceries and provides you with new recipes to try out based on what you have in stock. The machines will be so smart that they will know who they are interacting with and can tailor their behavior based on the individual needs of each user. The appliances will also be aware of what is happening within the context of the kitchen and will interact with the right information at the right time. There are all kinds of ideas you have for new product features, and there are so many ways for the appliances to interact with people. Your opportunities are boundless. Your users will be delighted with their experience.

But wait, what does it mean to have an interactive and intelligent kitchen appliance? What should an interactive coffee machine do? How should your fridge behave when it is communicating with you? What needs or desires does an intelligent fridge address for the user? How do we make sure that the interaction between the user and the product is helpful rather than controlling? How are you going to *design* the interactions of these appliances so that they fit into the social context of people's lives? How can you *learn* more about the environments these products will be deployed in, the uses people will want, and the problems people will encounter? These questions form the central motivation of this thesis.

The Internet of Things has expanded beyond industrial settings to encompass everyday products from toothbrushes to autonomous cars. Cheap microprocessors and wireless networking allow designers to make everyday objects “smart,” with the capabilities to collect data, make decisions, and interact with people. But, no matter how much computing power we can stuff into our products or how “intelligent” they might become, designers will always be tasked with creating products that are useful, usable, and meet some need in a person’s life.

Throughout the 20^{th} century, designers have crafted various human-centered design methods to help us understand our users and uncover their needs. This process of finding needs, or *needfinding*, comprises various thoughtful activities to help designers explore and understand people in relation to the design of new products (Faste, 1987; Patnaik and Becker, 1999). Some methods include conducting personal observations of users (Faste, 1987; Kuniavsky, 2003), contextual inquiry around products (Holtzblatt and Beyer, 2014), and market research (Faste, 1987). Often, these methods involve collecting broad strokes of information about many people’s behavior or having direct one-on-one experiences with people in person. These methods have worked well for many practicing designers, but as our products become more computationally enabled, the landscape for designing interactions with products is changing. The code the determines interaction no longer relegated to desktops computers and mobile phones. New, computationally defined interactions are appearing in our homes, our workplace, and even our cars. These new contexts require that designers consider how these various contexts influence the interaction that people have with their products. As products become more adaptive and personalized for individual users, understanding this context becomes more important. Product ideas that work in the lab may fail for many reasons in the real world. We are now designing 21^{st} century products and services. Though our traditional design research methods have worked well, can we bring our needfinding methods into the 21^{st} century?

1.1 Motivation

As machines collect more data about their users, there have been efforts to develop ways for computers to observe and learn how to service the needs of their users. Some examples include the Lumière Project by Horvitz et al., which aimed to automatically identify a user’s goals and provide task support while using desktop office software (Horvitz et al., 1998); Chen and Cimino's use of clinical information system logs to identify patient-specific information needs (Chen and Cimino, 2003); and Rashid et al.'s "Getting to Know You" techniques for helping recommender systems learn about the preferences of new users (Rashid et al., 2002). These systems can allow machines to characterize users in limited settings automatically. I am interested in an alternative approach in which machine capabilities enable designers to perform needfinding in new ways. Central to this idea is the insight that data — and needs — do not automatically led to solutions; we still need designers to probe situations and synthesize the meaning of observations towards potential alternatives. While data-driven design may provide us a new lens, there is no replacement, as Henry Dreyfuss suggests, for field research to educate the designer to the needs of people (Dreyfuss, 1955). With new capabilities though, we can explore how designers might augment their needfinding abilities for the design of everyday products.

I am primarily interested in asking how designers might augment their perception of users by using the interactive devices that they are now creating. What if these interactive devices become tools for understanding the needs and experiences of people? In this thesis, I explore how designers can use interactive technologies as a way to do needfinding. I call this framework for doing machine-enabled needfinding the *Needfinding Machine*. Working with a needfinding machine allows designers to discover people's needs and understand user experience by allowing the designers to observe, communicate and interact with people *through* their products. The Needfinding Machine provides a "conversational infrastructure" (Dubberly et al., 2009) by which the designer can build their understanding of a person in an evolving fashion and the user's real context. The Needfinding Machine is not a machine that discovers needs on its own. Rather, the Needfinding Machine extends a designer's ability to perform traditional person-to-person needfinding by interacting with the user and observing the user experience *through* the machine. It is computer-mediated communication between the designer and user *under the guise* of the Internet of Things.

1.2 The Needfinding Machine[1]

Faste defines needfinding as an active process of perceiving the hidden needs of specific groups of people (Faste, 1987). He has outlined a non-exhaustive list of needfinding methods that designers can use to better understand people, including market-based assessments, technology pushes and forecasting, and personal observations and analyses. Patnaik and Becker further describe the designer's process of needfinding as an organized, qualitative research approach to support new product development (Patnaik and Becker, 1999). Needfinding processes have been adopted within much of human-centered design practice (Kelley, 2007). Within human-computer interaction, needfinding is often focused on developing user requirements to guide product development and usability (Beyer and Holtzblatt, 1997; Kane and Yuschik, 1986) and helping designers to develop empathy for their users (Wright and McCarthy, 2008).

The *Needfinding Machine*, then, is a method used by designers to further their efforts to understand user needs in relation to a specific context by conducting needfinding through an interactive device. When employing the Needfinding Machine method, an interactive device becomes *a needfinding machine*. A needfinding machine is a physical instantiation of the Needfinding Machine and often has a unique name, such as DJ Bot. A needfinding machine is embedded in the user's environment and their everyday life. This setup allows the designer to explore distant environments, interact over large time scales, see and collect data, elicit information from the user, and prototype interaction in ways that overcome previous limitations of observational design research such as requiring the designer to be physically present, limiting the time of observation, and relying only on the designer's interpretation of the situation (Jones, 1970). Information from the user-product interaction is fed back to the remote designer who then uses this information to enact some control and alteration of the user-product experience. This way of working is inspired by Forlizzi and Battarbee's (Forlizzi and Battarbee, 2004) framework for understanding the experience of interactive systems. Like Forlizzi and Battarbee, the Needfinding Machine centers on user-product interaction as a way to understand user experience. Through the situated interaction within

[1] The following sections of this chapter are derived from Nikolas Martelaro and Wendy Ju, "The Needfinding Machine." *Social Internet of Things*. Springer, Cham, 2019. 51-84. (Martelaro and Ju, 2018)

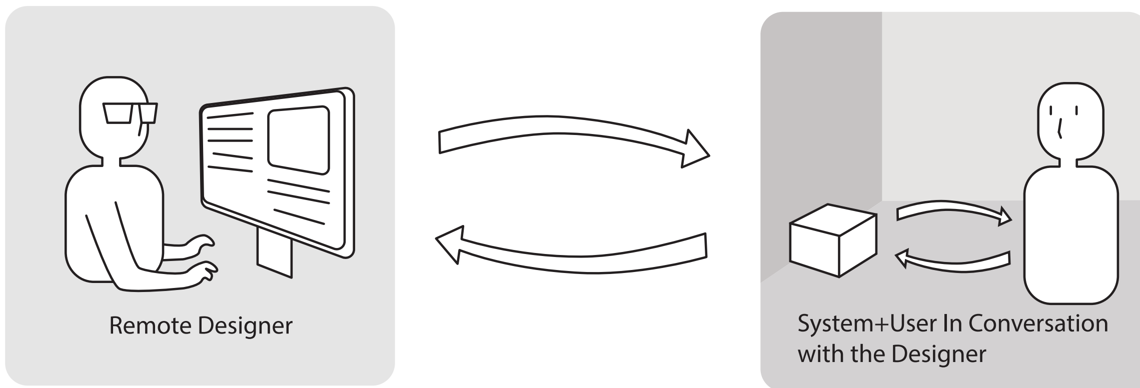


Figure 1.1: Information flows in a needfinding machine. The remote designer interacts through and performs an interactive system situated in the user's environment. The designer has a verbal conversation with the user mediated through the system (using text-to-speech) and controls aspects of the system's behavior. The designer observes and records these interactions to facilitate their needfinding efforts.

the real world, the designer can collect information about the user and their experience with the interactive device and the context. The information flows for a needfinding machine are shown in Figure 1.1.

During use, a needfinding machine provides designers with real-time access to objective system data (sensor readings, system logs) and qualitative observational data (video, audio). Moreover, it allows the designer to actively converse with the user through Wizard of Oz interfaces (voice, screens, tangible interfaces) (Dahlbäck et al., 1993). This conversational infrastructure (Dubberly and Pangaro, 2009) allows the designer, user, and the machine to interact in a situated manner (Suchman, 1987) towards the goal of understanding the user's needs in relation to a specific context. The Needfinding Machine considers an interactive device as a meeting point between the designer and the user (Simon, 1969). By allowing observation and interaction, the designer can use a needfinding machine to understand the user and take preliminary action towards satisfying the user's needs (Simon, 1969).

Consider our first thought exercise of being a designer tasked with creating a line of smart kitchen appliances. In this case, a designer creating a smart coffee machine might have a remote connection to the machine. This allows the designer to observe the user making their morning coffee. The designer could control the machine's speech, for example, testing out reading the morning news to the user. A live camera stream could provide the designer with the ability to recognize different people in the home, allowing them to try out personalized news stories for each user. The designer could also ask the user questions about their experience. *How are the new coffee beans you just got? What new food are you interested in trying out? What news do you want to start your day?* As the needfinding machine is embedded in the kitchen, the designer can also explore how the coffee machine should behave given the context of the kitchen. *What should the coffee machine ask when the user is sitting at the table versus cooking at the stove? How about the other appliances? How is the smart fridge doing with its recipe suggestions? Is there anything in your kitchen you'd like to automate?*

In addition to questions about what functions the coffee maker might have, a needfinding machine can help the designer understand how the machine should behave. This helps the designer understand the interaction needs of the user. *When should the machine talk? When should it alert the user that it needs to be cleaned? How should the machine communicate that it wants attention?* These questions can be iteratively explored through the designer's interactions with the user through the machine. In time, the findings from these interactions can be incorporated into the behavior of a final product.

1.2.1 Considerations for Needfinding Machines

Remotely accessing a user's environment and interacting with them through an interactive device can provide a designer with many potential ways of collecting data about the user. The technology that a needfinding machine is built on allows for both covert and overt interactions with the user. The use of such technology is dependent on the intention of the designer. While I cannot prescribe how a designer learns about their users through the user's interaction with an interactive device, I can suggest an intention for the Needfinding Machine method. Because the Needfinding Machine is intended to be a *conversation* between the designer and the user, we should follow good conversational etiquette. Through promoting good conversation, we can respect the user and obtain honest feedback on a design concept. Some considerations for using needfinding machines include:

Facilitate conversation — Needfinding machines help the designer build understanding through *interaction* rather than *surveillance*. This interaction is intended to be an overt conversation that builds a relationship between the designer and the user and is conducted with respect toward the user. Simply watching the user does not allow for the designer to ask questions and understand what the user is thinking and feeling. Additionally, if the user determines that they are being spied on, they will most likely close off any interaction they may have with a product or prototype. When used to facilitate a conversation, a needfinding machine allows for observation, action, and analyses simultaneously as a way for designers to explore unknown needs around a product (Hummels and Frens, 2008). By interacting with users through an artifact and by engaging the user in conversation, a needfinding machine can "amplify designer understanding of the intended purpose(s) of the artifact and may provide information that does not come out of initial interviews, observations, and needs analysis."(Abras et al., 2004)

Focus on the user — The goal of the Needfinding Machine is to aid the designer in developing an understanding of the user in context, not to justify the existence or usability of the machine in that context. To be a good conversational partner, we should avoid being self-centered and talking about ourselves too much. Learning from the user requires the designer to ask open questions that are focused on the user's experience. A machine that asks *"Do you like this?"* or *"How am I doing?"* can lead to overly polite responses from users (Nass et al., 1999). Just as a designer should not lead off needfinding by telling users what they plan to build or asking if the user likes a prototype (Kuniavsky, 2003), needfinding machines should focus on how the user feels and experiences the interaction rather than on validating their existence.

Focus on learning the user's needs — Even worse than being self-centered is being self-promoting. Designers can sometimes fall into the trap of trying to convince a user that their idea is what the user wants or that a product is good. It is possible for a designer to be too attached to their ideas or their preconceptions about a user. In this case, the designer may fall into the role of marketing a product rather than trying to understand the user genuinely. While practicing designers are always somewhat motivated to create a product that can be sold to the user, it is important for designers to work from user needs rather than try to push a product or interaction experience to a person. The intent of a needfinding machine should not be to persuade the user that an idea is good, nor should the machine try to convince a user that they have a need that is not truly present (but is advantageous for the sales of a product). Focusing the conversation on learning the user's needs rather than promoting a product is more likely to provide genuine value to the user.

1.3 What is in a needfinding machine?

In this section, I describe what elements are required to make an interactive device into a needfinding machine. I use a hypothetical Internet of Things coffee maker as an example device that designers can use to do needfinding work in a home environment. Specifically, we can imagine a design team tasked with understanding the user experience of a smart coffee maker as well as understanding the broader relationship that a user has with coffee and the kitchen.

The essential elements of the Needfinding Machine are functional blocks which support the user-machine interaction and the designer-machine interaction. These elements are shown in relation to the interaction loops in Figure 1.2. For the user-machine interaction, we build on Eric Dishman's formulation of design research where designers *observe*, *ask*, and *perform* in order to understand users (Dishman, 2003). A needfinding machine should allow the designer to *observe* the user in context, *ask* about the user's experience, and *perform* the machine's interactions with the user. I extend Dishman's elements of design research to include functions required in a needfinding machine for the designer-machine interaction. A needfinding machine should provide ways to *display* data about the machine and user, *control* the performance of the machine's interaction with the user, and *document* the observations that occur during the interaction.

Much of the technology used in a needfinding machine is similar to the technology used within the Internet of Things or "smart" products. By incorporating remote observation, data recording, and remote control interfaces, needfinding machines have capabilities designed for interaction prototyping and needfinding. To build needfinding machines, a design team must be able to build or augment an interactive technology with the required computing and communication technologies that enable interaction and communication between the designer, user, and product. In the following section, I describe the high-level capabilities in a needfinding machine. For a discussion of the underlying technical architecture that needfinding machines are built on, see Appendix A.

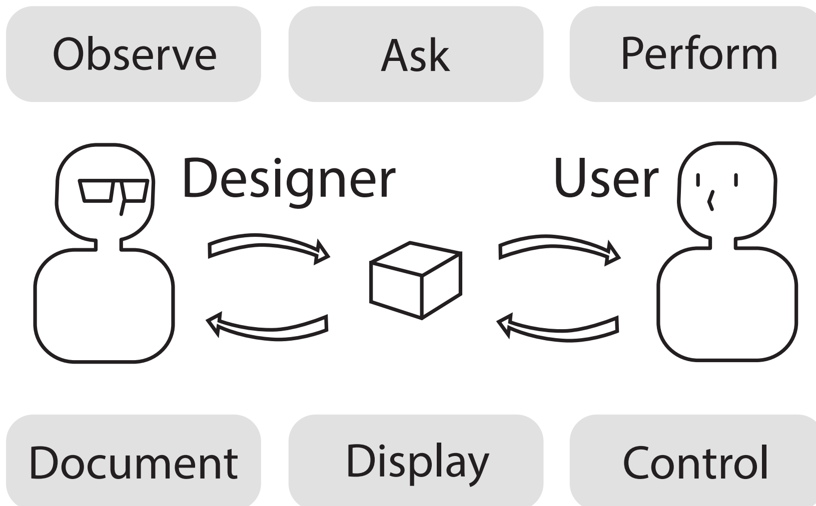


Figure 1.2: Functional elements of the Needfinding Machine in relation to the user and to the designer. The designer observes the user-product interaction, asks the user questions, and performs the machine in real-time. The Needfinding Machine helps the designer to document the interaction, display information, and control the interactive device's behavior remotely.

1.3.1 Observe

Observation allows the designer to see how users behave within a specific context and respond to different events. These observations can include both qualitative and quantitative information streams, depending on what the designer is looking to perceive.

Cameras and microphones can provide a high bandwidth picture of the user's immediate context and actions. Sensor and system data can show the designer information about the user's context that is often not directly observable *in situ*. This information is streamed back to the designer using a high-speed internet connection and displayed through various indicators and data visualizations.

The placement of the cameras and the selection of the data to be monitored by the designers is critical to consider; these decisions about what to instrument in the user's environment embody hypotheses on the part of the designer about what sort of information they might be looking for or need to support their interaction. For our Internet of Things coffee maker, we might put a camera facing into the kitchen that can see the user as they approach the machine and interact with any physical interfaces. The camera can also give the designer a view into the kitchen, allowing them to observe people's morning rituals and interactions with other kitchen objects. A microphone lets the remote designer hear the participant as they answer questions and talk about their morning experience. Buttons and knobs can be instrumented so that the remote designer can see how the user interacts with the machine and what settings the user changes.

1.3.2 Ask

By asking questions through the machine, designers can elicit information that cannot be observed, such as what the user thinks and feels. By asking the user questions, the designer establishes the interaction as a conversation, inviting the user to engage and participate in the needfinding process. These questions can be planned before an interaction. However, just as with any conversation, the appropriate questions for each situation are often revealed over the course of interaction with the user.

To enable question asking, a needfinding machine needs a communication interface. These questions could be text-based or spoken. Questions can also be asked without speech, such as through physical movement of the device which prompts a response from the user. While each of these modalities can be used by the designer to ask questions of the user, I have typically used speech-based interaction to ask questions. Specifically, I use text-to-speech systems so that the questions appear to come from the machine rather than directly from the designer. Text-to-speech allows the designer to have a conversation while still being removed from the situation and avoiding any biasing factors of their presence.

In the case of our Internet of Things coffee maker, we can use a text-to-speech system on the machine to ask the user questions about their coffee making experience such as *"What is important in a coffee machine?"* and *"How much customization would you like in a coffee machine?"* We can also ask broader questions about the user's relationship with coffee, such as *"What is the best part about drinking coffee?," "When did you first start drinking coffee?,"* and *"What would life be like without coffee?"* Furthermore, the designer can also ask about the rest of the user experience in the kitchen, for example, asking questions about the microwave and fridge, or what type of cooking the user likes to do. Using text-to-speech allows a needfinding machine to maintain its machine alibi, and aids in creating a consistent voice and persona around the user's interaction with the machine over time. Using machine voice also keeps the interaction situated in an Internet of Things context, making a discussion about other things in the kitchen somewhat plausible.

1.3.3 Perform

Interacting through the machine allows the designer to *perform* as the machine. This performance allows the designer to explore potential interaction opportunities and use physical or digital interactions as a means for eliciting needs from the user. In addition, the designer can also explore the machine interfaces themselves, giving them a sense of the machine's needs and limitations concerning potential design ideas.

Depending on the specific context, the designer can "perform as the machine" in various ways. This performance may include tangible, graphical, or auditory interfaces. It may also include interactions with other devices in the environment such as phones or other Internet of Things products. Each interaction that the remote designer can perform represents a degree of freedom that the designer can experiment with throughout their interaction. This may require the designer to build rapid functional prototypes of an interactive system. However, commercially available products can also be re-purposed for needfinding. For example, technology such as VNC or TeamViewer can enable remote control of graphical user interfaces.

In our coffee maker example, the designer might augment a commercially available coffee maker with smart capabilities. The designer can perform various functions of the coffee maker, such as setting the coffee preference per user or controlling when the coffee is made each morning. The designer can also explore new functionalities that a future coffee maker might have, such as providing the user with their morning news update, adding coffee to the user's shopping list when they are out, or even starting up the user's car once their coffee is ready to go. By performing as the machine, the designer can explore functionality that is not yet available. The designer can also test new interaction dynamics between the user and the machine, helping them determine what how the machine ought to interact and what technology may be required to enable new machine behaviors.

1.3.4 Document

By capturing the interactions with a needfinding machine, we can perform post-analysis and revisit our observations made during the live interaction. Capturing the action in the user's context allows the designer to look back and review the user's experience. Recording made in the user's environment also allow the designer to focus on contextual factors that they may have missed in-the-moment. Documenting the designer's environment can help the designer to reflect upon their actions during the session. Recordings of the designer's actions can be useful for developing interaction strategies for future interaction sessions and for making explicit the process that the designer is taking as they move through the design space.

Documentation can include recording video, audio, and data streams from the session. By recording the designer's control interface and any conversation they may be having with other designers, the Needfinding Machine can capture important moments that can reveal the designer's thinking during the interaction. Special interfaces such as pass-through audio/video recording devices, web-based data logs, and devices with built-in logging all contribute to the documentation of Needfinding Machine interactions. Our Internet of Things coffee maker can record video and audio from the user's kitchen and log button presses, coffee levels, or voice commands from the user. On the remote designer's side, we can keep a log of every question that was asked and interface that was controlled. We can also record what the designer sees on their screen and any conversation they might have with other designers participating in the session. After the session, these data streams can be synchronized for later viewing and analysis by the design team.

1.3.5 Display

Displaying video, audio, and data streams coming from the user's environment in real-time enables the designer to interact in real-time with the user. The display also supports the designer's observation and allows them take action on any data that may be relevant during their interaction session. Relevant information may in include video and audio from the user's environment, state changes in the interactive system, and time series information of certain product features. Often, the designer is presented with more information than they would naturally be able to see during an in-session interaction. For example, multiple camera views and live data from the machine are usually hidden to the user. When creating the display interface, the designer should take into account what they need to see and what aspects of the data might be interesting.

The display interface for our coffee maker might include a video window and a data dashboard. The designer might have one screen dedicated to the live video feed from the user's kitchen and another screen with the data dashboard. The dashboard might include live displays of the system settings such as supply levels or coffee temperature. If the designer is testing voice interaction, there could also be a running text log of what the coffee maker hears and interprets from the user. Each project will have different informational needs that require different configurations of data display. By taking the time to consider what information is pertinent to the interaction with the user, designers can better manage their attention during a live session. Just as important, considering what information should be hidden from live display can help prevent designers from being overwhelmed with data.

1.3.6 Control

A control interface allows the designer to perform as the machine and ask questions through the machine. Different machine behaviors require different control interfaces. For each element that the designer wishes to perform, a corresponding controller can be added to the designer's remote interface. Some of the most common interfaces include text areas for asking questions, controls for sending scripted speech, and controls for actuating physical interfaces.

For our coffee maker, the interface can have a list of questions or news stories that have been scripted for the interaction and a message box for sending custom messages that the designers create in-the-moment. Graphical toggle switches can turn elements of the coffee maker's graphical display on and off. Buttons can be used to send messages to other devices in the environment, such as the users' phone or to control something on the users' Internet of Things enabled car.

With a high number of degrees-of-freedom in a needfinding machine, the job of observing and interacting can become overwhelming. Depending on the rate of interaction, controlling the machine may require two or more people. With our coffee maker, it may be best for one remote wizard to control the speech, while another controls the physical interfaces on the machine or helps look up information like news to tell the user. By creating the control interface so that it can be easily accessed by multiple designers or easily replicated, designers can collaborate with each other and work from different locations. I typically use web-based technologies to create display and control interfaces so that all members of a design session can participate from any location by visiting a single interface.

1.4 Why Needfinding Machines

The purpose of the Needfinding Machine is to extend the designer's gaze and reach (Jones, 1970) by allowing them to see and understand user interaction in real-world contexts. Working through a needfinding machine can let designers engage people beyond themselves and their immediate community when working on the design of new technology products. Consideration and awareness of people who are different from the design team give designers a more informed position about the technology they are developing. While needfinding, understanding the experience of more people who are further from the design team can lead to designs with further reach and more impact on people's everyday lives. Furthermore, understanding and designing for more people provides an economic benefit by addressing a broader customer base.

A needfinding machine also helps designers explore new technologies as tools for connecting with users in the real world and crafting product interactions contingently. The Needfinding Machine framework takes advantage of several concurrent trends in technology:

- **Embedded computing:** Imbues everyday objects with computation, sensors, and network communications (Weiser, 1995). Allows devices to communicate with the Internet of Things and provides a way for designers to collect data remotely.

- **Cloud services:** Allow software and hardware to communicate across the internet, store data on remote servers, and enable new interaction capabilities such as machine vision and speech.

- **Online machine learning:** Allows systems to continually learn and update their models of users from streaming data. Can be used to support intelligent interaction between the machine and the user.

- **Conversational agents:** Lets people use natural language to interact with their devices. Provides and way for designers to capture their user's thoughts and feelings about a product or interaction.

- **Adaptive interfaces:** Attempt to change based on the user's preferences. Designers can explore what personalizations may be useful and what information is needed from the user to enable this adaptivity.

By utilizing and considering these technologies, a needfinding machine works as a tool to help designers understand their users better. A needfinding machine also allows designers to understand the needs of the machine better. By interacting through the machine, the designer can assess what it is the machine will need to understand and what data to collect to adapt to the user. This interaction helps to expose the designer to the new material of interaction data and allows them to play with potential interaction possibilities that consider this information.

1.5 Approach

This dissertation explores a new method for conducting needfinding and understanding of user experience by allowing designers to use interactive devices as a means to observe users in their everyday context and to participate in and change product interaction dynamics in real-time. The thesis asks:

- What previous models provide insight into the development of this new design method?

- How can the method be realized with currently available technology?

- Will people engage with products in ways that are useful for designers?

- What does the method do for designers that was previously difficult or impossible?
- Does the method provide useful information to designers?
- Why might we do needfinding through objects rather than in person?

To develop the Needfinding Machine, I built upon various theoretical areas of research including Design Research, User Research Methods, Cybernetics, and Interaction Theory. To realize the Needfinding Machine in a physical and usable form, I employed methods such as Research-through-Design (Zimmerman et al., 2007) and interactive device design (Igoe, 2014). To understand if people will engage in conversation with interactive devices, I performed lab tests to see how we may be able to use conversation between users and machines as a way for designers to understand people and their experience with a product. Finally, I conducted two real-world field deployments of needfinding machines done in collaboration with practicing designers to see how the Needfinding Machine extended designers perception and provided them with information beyond traditional needfinding methods. Each of these components helps to answer the questions of why we might do needfinding through interactive objects.

1.5.1 Point of View

This dissertation builds upon the idea that interaction with and observations of users can lead the designer to developing insights about the user's needs and wants (Dreyfuss, 1955; Faste, 1987; Kuniavsky, 2003; Patnaik and Becker, 1999). Any method that supports understanding users can be a way for designers to understand user needs. What the Needfinding Machine aims to do is extend the perceptual reach of designers (Jones, 1970) in ways that have not been previously possible.

One metaphor for thinking about the creation of the Needfinding Machine is comparing it to the development of a new space probe. To understand how well the probe is at supporting scientists in asking and answering their questions about distant planets, we must design, build and deploy the probe on a real mission. Though we can test certain components in the lab, the true test of the probe's ability to increase the perception of scientists is to see how well it performs in the real world. Similarly, while we can develop new design methods in a lab environment, the true test of any method is how it is used by practicing designers. In total, this work aims to present the concept of the Needfinding Machine, show how needfinding machines can be built, describe how they can be incorporated into design practice, and discuss how they increase the perceptual capabilities of designers.

1.6 Organization

This dissertation is presented in six chapters including this one. The remainder of the dissertation is organized as follows:

Chapter 2 situates the Needfinding Machine among the various theoretical areas that inform how designers understand people.

Chapter 3 presents a set of lab studies examining whether people will talk to their products if asked questions about their experience and describes various design features for promoting conversation between users and interactive devices.

Chapter 4 presents a real-world implementation of a needfinding machine in a car and describes case study with practicing automotive designers at Renault deploying a needfinding machine in order to understand the user experience of currently available advanced driving assistance features.

Chapter 5 develops a needfinding machine, called DJ Bot, for exploring music experience in context and presents a case study with interaction researchers at Spotify employing exploring the user experience of intelligent music systems in the car and the home.

Chapter 6 concludes this thesis by describing the benefits and limitations of using the Needfinding Machine along with a set of guidelines for deploying needfinding machines in practice.

1.7 Contribution

The work in this thesis explores how we can bring needfinding into the age of interactive devices. It builds upon well-established needfinding methods and proposes a way of using interactive devices as tools for understanding people. By leveraging the communication capabilities of interactive devices, designers can connect with users through interactive objects, allowing them to remotely observe and interact with them in the context of both the interactive object and the user's environment. The interactive device becomes a *conversational infrastructure* and a *feedback channel* allowing the designer to understand how the product plays into the user's life. The Needfinding Machine suggests how designers can extend upon traditional needfinding and interactive prototyping methods by situating themselves in the user's context during the moment of interaction. This method for needfinding can have implications for the development of future intelligent and context-sensitive products.

Chapter 2

Background

The Needfinding Machine builds upon existing theories and methods in design and extends the capabilities that designers have for understanding their users. This section presents the history of needfinding within design and then describes the system model for the Needfinding Machine. I then situate the Needfinding Machine among various frames for looking at interaction design including Design Research, Cybernetics, Interaction Theory, and Human-Computer Interaction and discuss how they inform the design of the Needfinding Machine.

2.1 A Theory of Needs

Needfinding is a process used by designers to explore and understand people's needs in relation to the design of new products (Faste, 1987; Patnaik and Becker, 1999). It is based on Maslow's theory of human motivation (Maslow, 1943) which presents a hierarchy of needs that people must meet to have a healthy and fulfilling life (Maslow, 1943). The base needs comprise *physiological needs* of food, water, and shelter and *safety needs* such as security. The needs then shift from a bodily to a *psychological* focus, with the next levels including relationships with others and feelings of accomplishment and esteem. The highest level of needs are *self-actualization* needs, whereby a person attempts to fulfill who they "*can* and *must* be." One can think of self-actualization needs as the need for an athlete to compete at the highest level or a designer who aims to create successful and delightful products.

From a design perspective, the hierarchy of needs serves as a model for the motivations that led people to use certain products and services. These fundamental needs push people to behave in specific ways to meet their needs. However, it does not make any prescriptions as to *how* the needs should be met. It is the job of designers to understand what needs are unmet and to design products and services that can fulfill these needs. Considering needs can be used as a starting point for designers to consider the creation of new products and services that alter a person's behavior and help them fulfill something that is lacking in their life.

Maslow suggests that when the lower needs are met, people will move on to fulfilling higher needs. When one is safe and fed, they will seek out intimacy and esteem. While physiological needs must be met for a person to live, psychological needs may be more or less important based on individual differences. For example, one may seek out self-actualization before seeking intimate relationship during specific times in their life (Tay and Diener, 2011).

While the hierarchy of needs is generalized, all people do not have all the outlined needs at the same time, nor does the hierarchy always hold. Individuals have their own set of needs that are more or less important based on many factors including the environment they are in and their internal motivations (Wahba and Bridwell, 1976). For designers, this suggests that understanding both the environmental and personal contexts of individuals is important for finding needs.

Maslow does consider the person's environment as an essential aspect of one's motivations but contends that it is not the only reason for motivation. He suggests that needs do not exist outside of the environment but that they are also not caused entirely by the environment. Internal motivations lead to behaviors which cannot be fully explained by context alone. This discussion of internal motivations aligns well with Weiner's teleological and goal-oriented view of behavior whereby people (and systems in general) possess internal motivators that led them to act and respond to changes in their environment (Wiener, 1948). The consideration of internal motivators also contrasts with Simon's "simple system/complex environment" (Simon, 1969) view of behavior. Both the environment and people are complex.

When considering an individual's needs, designers should consider how people's internal motivations interact with the environment they are within. When considering any design solution, we can use Maslow's description of needs as a grounding theory that we augment with an understanding of the individual and the context, taking into consideration the societal, environmental, and personal motivators that someone may have (King-Hill, 2015). This, in turn, motivates the designer to seek a holistic understanding of the user in their environment.

Maslow suggests that any unmet needs can cause "sickness" in a person. Of course, someone who is lacking proper nourishment would be regarded as unhealthy. However, it is interesting to consider that lack of esteem or actualization when one desires it can cause individual pain. While some needs may be considered more basic, any unmet need can cause pain and is worthy of fulfilling. Need is need. Faste posits that within design, any human requirement can be considered worthy for designers to address through their work (Faste, 1987). Needs can include people's wants and desires or their requirements to complete a task. More importantly, needs can focus on simple things that make daily interactions smooth and help people to achieve the higher order needs that drive their behavior.

With our grounding in what humans needs are and why designers are motivated to develop products and services that fulfill needs, let us now look at how designers *find* the needs that they can design for.

2.2 Finding Needs

To develop products and services that fulfill the needs of people, the designer must seek out and define what needs are unmet for a person or group of persons. Stanford's Robert McKim created the term needfinding as a way to describe the early stages of a design process where the designer's goal is to define new problems to work on rather than focus on solutions (Patnaik and Becker, 1999). The goal in this early stage of design is to understand a set of needs as "human requirements" (Faste, 1987) that can help designers generate various ideas for solutions. The motivation to find underlying problems rather than jump straight into solutions is that "needs last longer than any specific solution" (Patnaik and Becker, 1999). For example, we all need shelter, but given different contexts, the need for shelter can lead to many possible solutions. In our everyday life, the solution may be a house or an apartment. In the context of camping, a tent may be more appropriate. While traveling, a hotel or a friend's couch may suffice. Each of these satisfies the same need but takes into consideration the various contexts which that need may exist.

While one can begin with Maslow's basic categories of needs, more information about the situation and the individual is required for a designer to begin generating focused solutions. This brings up an important aspect of needfinding, and designed solutions in general, that context is a critical element for understanding and defining the needs and requirements that someone may have. With this in mind, designers have considered in-context needfinding as one crucial part of a full design process. As such, designers have developed theories and methods for the act of discovering people's needs.

Faste defines *needfinding* as an active process of *perceiving* the hidden needs of specific groups of people (Faste, 1987). Critical to Faste's view on needfinding are his remarks on perception. He suggests that finding needs requires the designer to somehow "see" unmet needs, however, this poses a conundrum. How can one see what is not obviously there on the surface? Faste argues that needfinding is "a paradoxical activity where what is sought is a circumstance where something is missing." Thus, a designer who is interested in finding needs is attempting to see something that is not currently visible. This requires designers to observe and interpret in order to formulate a statement about the unmet needs a group of people may have within a particular context.

Faste highlights explicitly that human perception is not a photographic representation of the world around us, but a reconstruction of the world within our mind. We reconstruct what we see and what we hear based on new input from the environment and previous images and knowledge that we have stored (Neisser, 2014). This act of construction thus requires some creativity on the part of the perceiver. Faste suggests that perceiving needs is akin to a creative problem-solving activity whereby needs are generated by the designer based on what they see and interpret based on their understanding of a situation. In this way, needfinding is a feedback-based activity; a designer seeks out information in the world, considers this information about their previous knowledge, and then updates their understanding to generate a statement about the needs of people. As a purposeful activity, designers have considered their process while attempting to find the needs of people and have developed guidelines for conducting structured needfinding.

Faste describes a non-exhaustive list of needfinding methods that designers can use to better understand people, including market-based assessments, technology pushes, forecasting, and personal observations and analyses. While many methods can be used, each method requires that the designer be open to seeking out needs that have not been met (Faste, 1987). Patnaik and Becker (Patnaik and Becker, 1999) further describe the designer's process of needfinding as "an organized, qualitative research approach to support new product development."

2.2.1 When do we do needfinding?

Structured needfinding activities typically during the early stages of the design process. During early the early stages of design, the designer is often concerned with *What is the right thing to do?*. Later, the designer is concerned with *What the right way to do it?* (Grudin et al., 1987). This process is well-described by the double-diamond model of design (Design Council, 2005), shown in Figure 2.1. Many needfinding efforts often occur during the first phases of the design process, where designers are going out and searching for needs. As they generate needs through observation of user, they will begin to converge on a specific needs which they would like to address.

After converging on a need, the designer then works to develop a product that satisfies this need. This leads to another generative phase often composed of prototyping and rapid testing. Needfinding also occurs during this phase of the design process. Creating prototypes and testing them with people often validates or invalidates previously found needs. Entirely new needs can also be discovered based on a person's interaction with a prototype. Furthermore, product-oriented needs such as information requirements or usability concerns can arise during the prototyping process. The interaction that people have with a prototype can elicit many new needs.

Finally, there is a third area where needfinding can occur, which is when a product is shipped. For most physical products, designers can learn about how their products are being used and then keep notes for their next development cycle. This often means 12–18 months between product updates. Nowadays, designers often learn about how people use their products through feedback postcards, online reviews, or Twitter comments. However, with software products, designers can continuously log data about product usage. They can

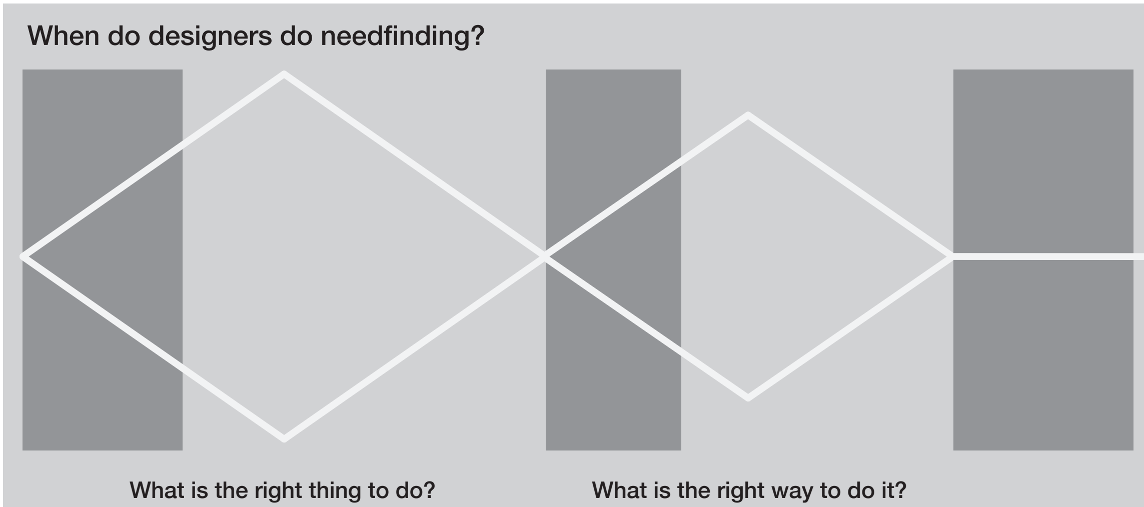


Figure 2.1: Needfinding during the double-diamond design process Design Council (2005). The gray bars represent when designers typically do structured needfinding. Needfinding often occurs during the early generative stages of trying to figure out what it the right thing to do and during generative prototyping phases when designer try to figure out how to develop a product right. Today, with computational capabilities, needfinding can also occur after a product is shipped.

continuously collect feedback from users and can more rapidly update their products as new needs are determined. As physical products become more computationally enabled, there is an opportunity for product designers to co-opt some methods from software products to learn about and update their products continuously. This is especially true for interactions between people and computationally enabled products where interaction models can be updated with software alone.

My goal in creating the Needfinding Machine is to provide a structure for designers to do needfinding through computationally enabled interactive devices. Needfinding machines can be employed throughout the design process. During early stage design work, a needfinding machine can be used for broad in-context explorations. As a product concept develops, interactive device prototypes can also be needfinding machines, especially around product interactions. Finally, once a product is shipped, it can retain its needfinding machine capabilities. This can allow the designer to stay connected to the user continuously. This connection can help the designer understand how the user's needs might change over time or what other needs may be unmet, thus restarting the design process.

2.3 Situating the Needfinding Machine

The development of the Needfinding Machine is inspired by prior work on design methods and interaction theory. This section provides an overview of four primary areas that have influenced the Needfinding Machine: design research, user experience research methods, cybernetics, and interaction theory.

The section on design research provides an overview of how we have come to understand the ways designers work and how this has been useful for the development of new design methods. I discuss various ways of thinking about design in both academia and within industry. I consider the Needfinding Machine as an extension of previous methods for understanding users. I discuss how the process of needfinding is a wicked problem and how this lends itself to developing *better* methods over *optimal* methods.

I then provide an overview of user experience research methods. These methods range from in-person interviewing techniques to data-driven approaches for understanding users. The various methods motivate many of the functional aspects of the Needfinding Machine.

The section on cybernetics discusses the theory of communication and control within intelligent systems. This provides a background on thinking about the ways that communication and feedback work to create systems that can learn and adapt to the needs of a system. Second-order cybernetics is discussed and related to the Needfinding Machine, which can be characterized as a cybernetic system for understanding users.

Building upon the discussion of cybernetics, I then conclude by looking at theories of human-human interaction. These theories are used as a foundation for understanding human-machine interaction and describe different ways of using interaction to understand the world. Because the Needfinding Machine facilitates conversation between designers and users, much of the work within this section is devoted to understanding the conversations that people have with one another within situated environments.

2.3.1 Design Research

Design research is interested in understanding both the products and the process of design. Throughout the years, there have been many forms of research in and around design that have evolved as the process of design has evolved. The study of design process is of particular interest to the development of the Needfinding Machine as it motivates how we structure a method for helping designers complete design work.

Before the industrial revolution, design was predominantly craft evolution, whereby the designer would constructively create products by hand without drawn plans (Jones, 1970). Trial and error development characterized this form of design work without any representation of thought processes or decisions (Jones, 1970). Craft evolution was exceptionally good for meeting the needs of individual users (Jones, 1970; Pulos, 1983). As Pulos describes, the shoemaker or silversmith would meet early and often with the customer, determining their desires and requirements before developing their goods. The craftsperson would then coordinate with the customer to adjust the product as needed to ensure proper fit and alignment with their desires (Pulos, 1983). Craft evolution was directly concerned with and explicitly tied to understanding and satisfy the needs of individuals. While craft evolution can lead to well-balanced products with exceptional fit to an individual (Jones, 1970), it does not scale well to meet the needs of many people.

With the rise of industrialization, the design and manufacture of goods could reach more people. Mass manufacturing would allow for hundreds or thousands of products to be made at an affordable price. This extended the reach of designers and their ability to satisfy people's needs. However, it required that products be identical; everyone would receive one version of a product rather than a product that was tailored to themselves. This also required a new structure for designing, specifically, design by drawing (Jones, 1970). By drawing out the plans for a design, the designer made their thinking more explicit, and thus, reproducible by the manufacturer. This also had the benefit of making the design process itself as thoughts are captured on paper and easily shareable. This documentation allowed more stakeholders to engage in the design process. This allowed different perspectives on a design challenge as well as more representation of different stakeholder needs. The production of industrial goods also required more coordinated labor and supply logistics. With more stakeholders involved, planning, decision-making, and explicit documentation became an integral part of the design process (Jones, 1970).

Following the beginning of industrial design, the design methods revolution of the 1960's saw the rise of various ways to make the design process more explicit and objective. These new methods were best captured in John Chris Jones' seminal book *Design Methods* (Jones, 1970). Many of the described methods attempt to make design into a scientific and objective process that could be repeatable across different product domains and scales. However, the scientific aspects of design are not considered to be the same as the natural sciences. Rather, design is interested in the "sciences of the artificial" or the sciences of human-made systems, as described by Herbert Simon (1969). Simon considered physical goods, urban spaces, financial systems, and social organizations to all be designed systems.

Designed systems can be characterized by their complexity. Specifically, many human-made systems are considered to be what Weaver describes as *organized complexity* (Weaver, 1947) where many known and unknown interacting variables influence the system's behavior. Organized complexity comprises a class of problems that can not be solved easily with explicit methods used for simple one- or two-variable problems. Organized complexity also cannot be described using purely statistical means which are useful for disorganized complexity such as particle-based systems (Weaver, 1947). Thinkers such as Fuller and Simon considered design as one way to address systems of organized complexity. Their ideas further developed into the areas of *systems thinking* and *design science* (Fuller, 1968; Simon, 1969).

The proliferation of digital computers further fueled the development of design methods that sought to be more objective. Simon outlined how the computer could be used to create processes for design that could help to find optimal solutions through an iterative search of a problem space (Simon, 1969). Others, such as Papalambros and Wilde developed models for optimal design in specific areas of engineering design (Papalambros and Wilde, 1988). These methods were particularly useful for the design of flow systems (Jones, 1970) where known components with fixed input and outputs could be put together to meet a well-defined need. For example, the design of most air conditioning systems can now be automated with the proper constraints and a set of components and functions that can be strung together.

While optimization methods can solve constrained design problems, they cannot address the complexities of many systems, especially those which involve human interaction of some sort. Indeed, Rittel and Webber helped to define the challenges of designing socio-technical systems through their conceptualization of *wicked problems*, or problems which inherently have no right or wrong answer and cannot be optimized because there are no definitive optimizations (Rittel and Webber, 1973). Rittel and Webber argue that science alone was not set up to solve these problems, suggesting that designers could create solutions that may lead to *better* outcomes given a specific situation but that they could never truly optimize these solutions for all peoples or all contexts (Rittel and Webber, 1973).

Many of the interesting problems in design today are considered to be wicked problems. Solutions cannot be optimized using purely scientific principles. Additionally, the development of design processes themselves are viewed as wicked problems (Buchanan, 1992). To date, there is no optimal way of approaching a design problem. Instead, designers are tasked with assessing the design situation, choosing an approach to forward their thinking and their product development, evaluating the outcomes of their decision, and then reflecting on their actions so that they can iterate on their design; what Schön describes as a *reflective practice* (Schön, 1984).

Even within a design process, the individual activities that designers do cannot be optimized. This is especially true for the process of understanding users and finding needs. It is not possible to optimize the process of needfinding, nor to develop a method that can work across all design domains. This being said, we can develop systems for supporting designers in understanding people's experience. We can also asses how designers use these methods to do design work. Insights from observing designers (Tang, 1991) can be used to help develop new ways of needfinding that can be appropriate during some design scenarios.

In some ways, the Needfinding Machine approach returns design to its original craft evolution roots by championing the iterative and contextually situated nature of solving design problems. However, it does not give up on the idea of making the thoughts, plans, and decisions that occur during a design process explicit. Indeed, Jones suggested in 1970 that new design methods would most likely incorporate aspects of craft evolution, systems thinking, and optimization. In creating the Needfinding Machine, I aim to develop a system that provides designers with an approach to needfinding that makes the designer's process explicit and allows designers to iteratively evolve an understanding of users based on actions and feedback that occur in the real world.

2.3.2 User Research Methods

In this section, I review methods that have been used by designers to help understand users. For each method, I provide a brief overview of its use in design and discuss which functional components from Section 1.3 are incorporated into the Needfinding Machine.

Ethnography

Ethnographic research is the foundation for much of what is considered design research in practice. Within many design contexts, practitioners act as participant observers, embedding themselves within a context to understand people. This tradition arises from Geertz's "thick description" of people and their behaviors and situates the observer as having a specific point of view that allows for specific interpretation of people's actions and motivations (Geertz, 1973). For example, when users quickly change a song on the radio, are they interested in listening to something else or does that song harbor undesired meaning and emotion?

This process of interpretive, contextually situated ethnography has translated well to design work and allows the designer to observe the lives and experiences of their users. However, most companies do not perform academic ethnography (Ladner, 2012), which can often take months or years of intensive study. Rather, designers have adapted ethnographic methods into short, focused participant observations often lasting on the order of days (Millen, 2000; Plowman, 2003). Even with short observations, ethnography-inspired methods have become staples for finding user needs and supporting generative design activities (Kelley, 2007; Laurel, 2003; Sas et al., 2014).

Within human-computer interaction, ethnographies are often required to report on some *implications for design*. Though Dourish argues that requiring design implications of academic ethnographies can undermine the richness of these studies (Dourish, 2006), interactions through a needfinding machine are specifically situated to support design work and thus help designers generate implications for future design. Additionally, designers using a needfinding machine are interested in understanding user needs in relation to a specific product or context. While designers learn about the broad aspects of user's lives, the designer's performance as the machine grounds needfinding around the user-product interaction.

Things as Co-ethnographers

As the Internet of Things becomes an everyday reality within people's homes, there is growing interest in how designers can use information from the viewpoint of things to understand and empathize with people in context. Projects such as Comber et al.'s BinCam (2013) and Ganglbauer et al.'s FridgeCam (2013) used cameras attached to products to collect images of everyday interactions. By collecting pictures and videos from the point of view of objects, the research teams could observe aspects of people's lives that would usually be out of view during interviews and short observations. After using similar methods of collecting pictures from cameras placed onto everyday objects, Giaccardi et al. have suggested that the software and sensors of IoT objects can give designers access to "fields, data and perspectives that we as human ethnographers do not have, and therefore may help us to 'see' what was previously invisible to humans." By providing a different viewpoint, the things become "co-ethnographers," working in conversation with the designer to help them understand the user from a different and situated perspective (Giaccardi et al., 2016a,b). Wakkary et al. extend this idea of thing-centered understanding of people to focus primarily on the relationship between things rather than focusing on direct observation of people. Their work explores how focusing design inquiry on things and their interactions can inform the relationship that people have with internet connected products. For example, people attributed human-like qualities and an ability to identify people in the home to a set of plates and bowls that communicated invisibly with each other and on Twitter (Wakkary et al., 2017).

The Needfinding Machine is related to the use of things as co-ethnographers. However, human designers remain in the interaction loop with users with a needfinding machine. While things as co-ethnographers allow designers to observe and document people's interactions with things, they do not provide the designer the ability to control the machine's performance or view real-time data about an interaction. Moreover, by acting as the machine, the designer can gain an understanding of interaction challenges the machine will face. By mediating their interactions through the machine, the designer can reveal both the needs of the user as they interact with the technology and the needs of the machine as it interacts with a person.

Remote usability testing

With the rise of high-speed internet and mobile devices, designers are now able to explore user experience remotely. More traditional usability testing methods have been modified to be performed remotely so that the designer does not need to physically "be there" to build understanding about the user and product (Black and Abrams, 2017). Waterson et al. (2002) and Burzacca and Patern (2013) each test the usability of mobile websites by collecting data from people using the website on devices outside the lab. Often, these methods have been created to explore the use of mobile devices beyond traditional lab environments. For example, English et al. (2004) conducted remote contextual inquiry to improve enterprise software, and Dray and Siegel (2004) used remote usability testing to explore international use cases for their software. Depending on the study setup, remote usability testing can be done synchronously, where the researcher is observing the remote activity as it is happening, or asynchronously, where the researcher is analyzing data logs or recordings at a different time (Black and Abrams, 2017).

Although being out of the lab can reduce study control and be more challenging for data recording, Brush et al. (2004) and Andreasen et al. (2007) have found that synchronous remote methods can be just as effective as being present with the user. In addition, remote interaction and observation can reduce the pressure participants can feel from having a researcher continually looking over their shoulder (Andreasen et al., 2007).

The Needfinding Machine is inspired by the kinds of observation and documentation that remote usability testing provides designers. Similar to remote usability testing, a needfinding machine enables designers to observe and engage with remote users synchronously. The Needfinding Machine also documents data from the interaction similarly to remote usability testing. However, Needfinding Machines differ from remote usability testing as designers engage with the user by performing as the machine rather than being on a phone call with the user as they are trying an application. The ability for a designer to perform the machine positions a needfinding machine earlier in the design process and focuses the designer on learning by interacting with the user, not just through data collection. Furthermore, the intent of a needfinding machine is to help designers understand the broader needs of users, rather than focus on testing how usable a product is.

Data-driven design validation

With devices generating more data, there is a growing interest in using data for understanding users. Geiger and Ribes (2011) use system logs to conduct ‘digital ethnography’ on users of online blogs and wikis and Christian (2012) outlines how websites test and refine new designs using A/B experiments. These methods provide a way for designers to observe how users engage with a product based on objective data measures. The use of objective data can help the designer avoid some challenges with direct observation such as researcher interpretation of events and participant bias due to the researcher’s presence (Patton, 1990). Data-driven methods also allow for large-scale observation, helping designers see a broad range of interactions that users have with an interactive system.

Still, many methods that rely solely on data logs can only show what a user is doing and only can see data from what is instrumented. Attempting to understand users only from interaction logs can run the risk of being too granular (if the data is too noisy) or too high-level (if too many data points are aggregated). Moreover, data logs cannot always tell the design how a user feels about their interaction with a product.

Some projects attempt to capture how people feel through surveys. For example, Cadotte’s Push-Button Questionnaire for understanding hotel experiences collects information about people recent visits to a hotel chain (Cadotte, 1979). A modern version of these survey machines, called Happy-or-Not[1], simplifies a questionnaire into four simple smiley face emotions as a way to gauge customer satisfaction in public spaces such as airports and sports complexes (Owen, 2018). These survey kiosks allow for businesses to gather some level of satisfaction data quickly. For example, when many customers rate things negatively, a member of the business can go to the site and figure out what is wrong. This shows how small bits of focused emotion data can be used to understand some aspects of customer experience. Still, data-driven approaches often prove more appropriate for design validation and optimization rather than generating new design ideas. While data-driven design can be useful for beta testing usability or optimizing the experience of a particular location, designers are often interested in why users are behaving in a certain way: what are their motivations, their goals, challenges, and thoughts?

[1] https://www.happy-or-not.com

Experience Sampling in the Wild

To help understand both the *what* and *why* of user experience, researchers have employed experience sampling methods on everyday products (Csikszentmihályi et al., 1977). For example, Consolvo and Walker used text messages on mobile phones to ask users question during their day (Consolvo and Walker, 2003) and Froehlich et al. blend interaction logs with randomly timed text message questions about the user's experience (Froehlich et al., 2007). Aldaz et al. used similar experience sampling questions through a phone application designed to help hearing aid users tune their hearing aid settings (Aldaz et al., 2013). By collecting user experiences while tuning their hearing aids, Aldaz et al. found that blending interaction data and the user's in-the-moment experience can allow for new forms of needfinding beyond in-person interviewing and observation.

While these experience sampling methods aim to elicit the user's experience with a product, they often focus on text-based descriptions of experience. Froehlich et al.'s My Experience system did provide images and video that people captured on their phones, helping researchers to understand the user's context better (Froehlich et al., 2007). However, these media clips were captured when users took them rather than when the researcher may have wanted to see an interaction. Crabtree et al. captured video clips from third person cameras while exploring Ubicomp games that blend online and real-world tasks (Crabtree et al., 2006). They then synchronized these clips with sensor readings and device logs to "make the invisible visible and reconcile the fragments to permit coherent description" of the player's experience. The Needfinding Machine builds upon Crabtree et al.'s insight that mixing video and data provides designers with a high-fidelity view of the user's experience. What is critical for a needfinding machine is the real-time *video* of the user's environment. The video allows the designer to observe the user's experience and allows them to inquire about the user's experience at the moment of interaction, rather than during post-analysis or through a random experience sample. Live video also allows the remote designer to control the interactive device effectively. This control allows the designer to move beyond observing programmed interaction and allows them to explore and prototype contingent interaction. Finally, the video provides a rich context for the data logs that are captured from the interactive device, providing documentation beyond click-streams and device logs.

Probes

The Needfinding Machine concept takes many inspirations from the development and use of probes in design and HCI. Gaver, Dunne, and Pacenti's Cultural Probes (1999) provide designers with a means to understand and empathize with geographically distant peoples. Cultural Probes, often consisting of postcards, cameras, and guided activities, aim to elicit contextual information from people and help designers build a textured and rich understanding of people's lives. Hutchinson et al.'s Technology Probes (2003) extend Cultural Probes to include the use of technology as an eliciting agent. These probes allow technologists to understand how new devices may fit into everyday life and inspire new potentials for computational products. Originally, probes were intended to be provocations for collecting stories about user's lives that would lead designers to reflect on their users and their role in the design process (Gaver et al., 1999). Even when using technology, Hutchinson et al. suggest that probes are not prototypes to be iteratively developed over time, but should focus on eliciting user engagement and opening up design spaces (Hutchinson et al., 2003). This being said, Boehner et al. describe how HCI researchers have expanded the use of probes to include diary studies, photo journals, longitudinal studies, and participatory design prototypes (Boehner et al., 2007). Boehner et al. also discuss how probes have expanded beyond their original goals of promoting reflection to help designers collect data and generate user requirements for future design ideas. Amin et al. used a probe during a participatory group exercise to help develop a set of four design requirements for mobile phone messaging (Amin et al., 2005). Kuiper-Hoyng and Beusmans (2004) and Gaye and Holmquist (2004) each use probes along with interviews to help user discuss their experiences in their home and city, respectively. The use of probes helped each group to refine and iterate on more specific design projects.

The Needfinding Machine builds from using probes as a way to understand user needs, but focus the designer on considering implications for more specific product ideas. Thus, Needfinding Machines exist somewhere in a space between probes and prototypes. While probes can help to document user experiences asynchronously, needfinding machines focus on helping designers observe and interact with users in real-time. Needfinding machines also aim to collect data from the remote environment and display this to the designer so that they can continuously change how they control the machine's behavior. By performing as the interactive device, designers can test specific interactions with users; however, these interactions are not intended solely for usability testing but the broad exploration of potential needs given the user's context. Needfinding machines do retain the goals of probes to help designers understand the user's experience and life more broadly. Furthermore, Needfinding Machines build upon the ability of probes to elicit textually rich information from people in contexts that would be otherwise unobservable. Information collected from these interactions is intended to be analyzed in holistic and interpretative manners but also include more actionable data about the user's experience with the interactive product. By allowing the designer to ask the user questions and perform the interactive product's behavior with the user in real-time, needfinding machines aim to collect what Mattelmäki and Battarbee (2002) call "inspiring signals" for developing empathy with the user and develop what Boehner et al. (2007) state is a "holistic understanding" of the user's experience with a product.

Wizard of Oz

Wizard of Oz methods have often been used in design to simulate technologies that are currently unavailable. This method uses the "wizard behind the curtain" metaphor as a way to control prototypes when the product's technology is unavailable or intractable at the time of experimentation. Prototypes such as Kelly et al.'s (1983) exploration of natural language understanding and Maulsby et al.'s (1993) simulation of multimodal interfaces show how designers can learn a great deal about their proposed designs before allocating significant resources to technical development (Dahlbäck et al., 1993). When performed early in the design process, Molin et al. (2004) suggest that Wizard of Oz experimentation can help to define user requirements and promote collaboration between designers and users.

Designers can also use Wizard of Oz experiences to gain insight into what a user is feeling and thinking during the moment of interaction. For example, Sirkin et al. used Wizard of Oz to control a simulator-based autonomous car while asking a driver questions about their experience (2016). The improvisational style of these interactions allowed the driver to experience a potential future for autonomous vehicles and allowed the designers to gain insight into how drivers would react and respond to the car's behavior. This playful style of Wizard of Oz interaction prototyping and inquiry provides a foundation for how designers can collaboratively work with people to explore new interaction potentials and reflect upon their current and future needs. Furthermore, Maulsby, Greenberg, and Mander found that one of the most important aspects of Wizard of Oz prototyping is that designers benefit by acting as wizards; seeing uncomfortable users and find product limitations while acting as the machine can help motivate further prototype iterations (1993).

The Needfinding Machine extends the capabilities of lab-based and controlled Wizard of Oz for use in real-world contexts. Designers interacting as the machine keeps many of the same aspects of control, performance, and documentation of lab-based Wizard of Oz studies. Needfinding Machines also use Sirkin et al.'s (2016) use of improvisational interviewing through the machine with the goal of helping the designer understand a user's lived experience and potential needs around the specific interaction that is being designed.

Conversational Agents

Conversation around the experience of products is a powerful tool for understanding and moving forward with design ideas. Dubberly and Pangaro (2009) describe how conversation between project stakeholders allows design teams to co-construct meaning, evolve their thinking, and ultimately take an agreed upon action in the world. This echoes Schön's conceptualization of design as a conversation between the designer and the design situation (Schön, 1984). With this in mind, designers can use machines that converse with users, or conversational agents, as tools for understanding user experience.

Although human conversations can be quite complicated, even simple questioning from a machine can elicit meaningful responses from people. By the mid-1960's, systems such as Weizenbaum's (1966) ELIZA teletype Rogerian psycho-therapist could use simple rules to engage people in deep conversation about themselves. As conversational agent technology is becoming more popular within contemporary product design, design teams are exploring how to use chatbots to inquire about user experience in the real world. For example, Boardman and Koo of IDEO have used Wizard of Oz controlled chatbots to prototype a fitness tracker application, a text-based call center for public benefits, and a mobile application for health care workers tracking Zika (Broadman and Koo, 2016). Using the chatbots to engage people in conversation, the designers on these product teams were able to continually develop empathy for the everyday lives of their users. The chatbot conversations helped the designers uncover needs around the services they were designing. For example, the need for users to track healthy and unhealthy activities in their day and the need for users to feel safe while asking health-related questions. Additionally, by acting as the chatbots, the design team engaged other project stakeholders in controlling the bots. This led to debate and reflection on what the product ought to do and how it ought to interact, helping the team to better understand their designer values and the needs of their stakeholders.

The Needfinding Machine is similar to the work of Boardman and Koo at IDEO who use conversational agents as a way for designers to understand user experience with interactive systems. While using a needfinding machine, the designer can have a rich conversation with the user as the user is interacting with the product. Additionally, using people's innate ability to converse allows more members of the design team, even those without specialized training in interaction design or user research, to engage a user while acting through the machine. In our work on Needfinding Machines, we have used voice-based conversation rather than text messaging. This allows for more fluid communication and lets users describe their experience and answer questions during the interaction instead of needing to switch to a mobile phone messaging application. By using voice instead of messaging, a designer can explore experiences in environments where a user might be preoccupied with other tasks, such as cooking in their kitchen or driving in their car. The Needfinding Machine also differs from using chatbots alone by providing live video and data feeds from the user's environment. Having live video and data allows designers to use context as a basis for their conversation with the user and frees the designer from relying only on what the user is saying to understand the user's experience.

2.3.3 Cybernetics

Cybernetics provides a way of thinking about feedback based systems and interactions between intelligent agents (Wiener, 1948; Pask, 1975). More importantly, cybernetic systems use information, communication, and control to achieve goals (Dubberly and Pangaro, 2007). Specifically, second-order cybernetics provides a framework for thinking about how designers can learn about their users and suggests how we can develop systems for supporting designers to learn about users. If we consider needfinding as a goal-oriented activity to facilitate learning through feedback between a designer and user, then cybernetics can provide a useful frame for developing our Needfinding Machine.

Originally described by Wiener, first-order cybernetic systems have simple goals which do not change. These systems then use feedback between their current state and their goal state to determine their behavior and help them reach the goal state (Wiener, 1948). The canonical example of a first-order cybernetic system is the governor which sets an engine's speed by measuring the output and adjusting the input force so that the output matches the desired speed (Maxwell, 1868). While first-order cybernetics provides a simple way to view feedback systems and takes into account the goal of a system, it is not sufficient for describing the process that designers use when doing needfinding, or doing design in general.

Second-order cybernetics provides a more appropriate framework for thinking about the way feedback works during a design process. Second order-cybernetic systems learn from the feedback they receive and can use their understanding of a situation to generate new goals (Pask, 1975). In this way, designers are second-order cybernetic systems (Dubberly and Pangaro, 2015). While needfinding, the designer observes the interactions of users in the real world and makes changes to the world as a way to learn about their user and their design. This situates the designer as a participant-observer (Pask, 1961).

Second-order systems that can update their goals based on feedback from the environment and feedback from the self align well with Schön's idea of the designer as a reflective practitioner (Schön, 1984). During reflective practice, the designer observes the current situation and then takes some action, perhaps altering a sketch or a three-dimensional model, and sees how this change alters their thinking. From here, the designer may learn something about the design situation that they had not previously noticed. They can then update their goals based on their new understanding of the problem at hand. Schön considers this process to be a conversation between the designer and the design situation. One can think about this in the way that artists may talk about how a piece in progress "speaks to them."

Within cybernetics, conversation is viewed as one result of two cybernetic systems interacting with each other (Pask, 1975; Dubberly et al., 2009). The goals of conversation are to accomplish tasks and to discuss and update goals. Dubberly and Pangaro consider design to have the same goals as cybernetics (Dubberly and Pangaro, 2015). Similar to Schön, they consider design as a process of conversation and thus a cybernetic process (Dubberly et al., 2009; Dubberly and Pangaro, 2015). Other researchers, such as Lawson, also consider design as a process of conversation (Lawson, 2006) and specifically highlight how the explicit conversations that designers have with other designers and project stakeholders constitute a particular form of design work (Lawson, 1994).

Working from the premise that design is a conversation, Dubberly and Pangaro suggest that designers should develop systems that facilitate conversation (Dubberly and Pangaro, 2009). By creating new forms of conversation between design teams and end-users, we can improve our abilities to generate feedback and improve our learning about a design situation. This can help us to update our design goals and generate solutions that form a better match to the needs of people. This leads to an infrastructure for conversation.

My goal in creating the Needfinding Machine is to develop a conversational infrastructure between designers and users. Within a more extensive design process, needfinding is already one area where conversation between designers and users is made explicit through interviewing and user observations. As products become more computerized and networked, they can allow for a remote communication link between the user and designer. This presents opportunities for new forms of conversation around design.

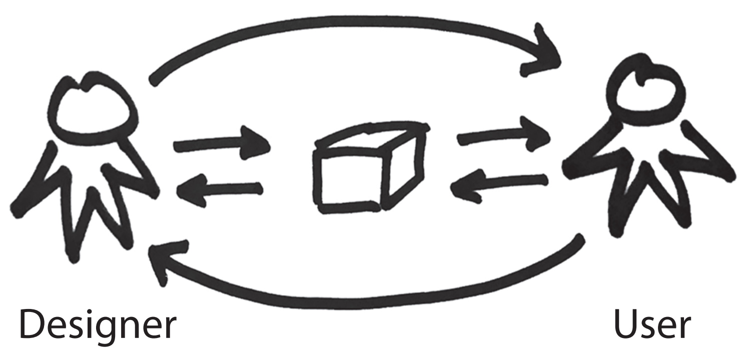


Figure 2.2: Feedback and feedforward communication loops during design exploration with a needfinding machine. The outer loop represents direct designer to user interaction through traditional needfinding methods. The inner loops represent interaction through a needfinding machine.

Figure 2.2 shows a simple diagram of the communication loops between a designer (on the left) and the user (on the right) through the interactive product (in the center). The outer loop is representative of more traditional needfinding methods that have the designer directly communicating to the user such as through interviews or in-person observations. The inner loops directed through the interactive product are new additions to the needfinding process and are the main contribution of this thesis. On the right side, the user-product interaction loop includes any actions that the product takes (upper line) and any data that is collected by the product (lower line). On the left side, the designer product interaction loop includes actions which the designer takes to control the product (upper line) and any data that the product feeds back to the designer (lower line).

2.3.4 Interaction Theory

Given that we consider design to be a conversation, we can look at different theories of conversation and interaction to help us develop the Needfinding Machine. These theories can motivate different attributes of the Needfinding Machine and describe how they can help designers better understand people. Building upon our discussion of cybernetics, Pask's conversation theory (Pask, 1975) describes how interacting agents use conversations in order to learn. In the model, knowledge is represented symbolically within the interactants. Conversation allows the interactants to make their knowledge explicit through language. By having a conversation, the interactants each provide feedback to each other, allowing the creation of a shared representation of their knowledge. The creation of shared representation between the designer and the user is a key goal of the Needfinding Machine.

Conversation theory shares many similarities with symbolic interactionism, which considers people's interactions as a primary force in shaping their behavior and places a value on the symbolic meaning of the objects and events people interact with (Mead, 1934). Interaction between people and objects is seen as a primary way to generate knowledge about people and about the social environment they are in. Within a symbolic interactionism framework, individuals are considered to be active, willful, and goal seeking and are influenced by the social context of which they are within. Furthermore, the internal symbolic meaning that people attribute to objects and environments is influential on the behaviors that they have. This aligns well with the internal motivations people have concerning Maslow's theory of needs (Maslow, 1943). These internal motivators also align well with cybernetic models of goal seeking agents that respond to their environment (Wiener, 1948). Additionally, the relationship between the internal and external environment echoes Simon's discussions on complex systems. Specifically, how complexity can arise from complex environments rather than complex internal states (Simon, 1969). This balance of understanding both the individual and the context is important for any designer doing needfinding. These theories suggest that to understand a person and their motivations, one must understand the environment that the person is in and also determine how the person relates to and gives meaning to that environment. Symbolic interactionism suggests that to learn about people, we must see their interactions within context.

The Needfinding Machine specifically grounds itself in using interaction as a primary mode of generating knowledge about a person. Furthermore, the Needfinding Machine approach to understanding people is intended to be situated within a user's environment. This allows the designer to observe the user's interactions in context and allows designers to ask questions which explore the symbolic meaning that people attribute to their context. The Needfinding Machine also uses interaction with the user via spoken conversation and remote control of objects as a way to elicit the needs that they may have.

Conversation and interaction can help designers build up a model of a person over time. Using a needfinding machine, the designer can elicit and document the internal motivations that a user has *through* their interaction and conversation with the user. When attempting to uncover hidden needs, the interaction between the designer and the user serves as an exercise in developing shared understanding. Both the designer and the user must come to an agreement about the user's needs. Further on in the design process, the designer and user must come to a shared agreement on an effective solution for the identified need.

To develop this shared understanding, the designer must develop a set of mutual knowledge, beliefs and assumptions with the user and a shared process for communicating—what Clark and Brennan define as *common ground* (Clark and Brennan, 1991). To develop common ground, two people consistently update their shared set of beliefs and assumptions through a process called grounding. Grounding is influenced by the communication medium that is being used. In the case of the Needfinding Machine, the channels which the designer employs for communication will determine how the user and designer interact and accomplish grounding. Clark and Brennen outline eight constraints of communication that impact grounding. We consider each constraint and its impacts on the designer's ability to perform grounding with the user through a needfinding machine.

1. **Copresence** – Are the conversation partners in the same space? The Needfinding Machine aims to give the designer a feeling of copresence with the user through various information channels. The user, however, should not feel copresent with the designer. This is done to remove the designer from the direct interaction and allow them to explore the interaction in the context of everyday experience rather than a supervised user test.

2. **Visibility** – Can the conversation partners see each other? We attempt to give the designer a full view of the user and their environment so that they can moderate their interaction based on the physical situation. This allows them to refer to objects in the user's environment during the conversation.

3. **Audibility** – Can the conversation partners hear each other? The remote designer using a needfinding machine can hear what the user is saying and can speak to the user through a speech agent in the interactive device.

4. **Cotemporality** – Can people hear what is said the moment it is produced? The Needfinding Machine is a real-time interaction. Only small delays occur based on the network transfer. One aspect that is not entirely cotemporal is that the remote designer must use text that is then translated into speech, meaning that they produce the message before it is spoken to the participant. This can slow the communication down and cause extra effort for the designer as they may not be fast enough at typing to send their message in time.

5. **Simultaneity** – Can both conversation partners communicate at the same time? A needfinding machine is not entirely simultaneous because the remote designer must type out their messages and then send them. However, information from the user is simultaneously received by the designer. This allows the designer to understand how their interaction through the machine impacts and influences the user's behavior.

6. **Sequentiality** – Can communication get out of sequence between conversation partners? A needfinding machine is effectively spoken conversation, and thus, messages will not be out of sequence.

7. **Reviewability** – Can conversation partners review each other's messages? Because most interaction is spoken, there is not a way for each conversation partner to see each other's messages in real-time. On the remote designer's side, this could be overcome with real-time speech-to-text transcription that would stay more permanently in the control interface.

8. **Reviseability** – Can conversation partners revise messages before they send them? The remote designer can review and revise messages before sending them to the user, however, the user cannot revise messages as they are speaking in-the-moment. This allows the remote designer to consider their communication with the user more carefully.

Overall, these interaction theories inform the design of the Needfinding Machine as a tool for facilitating conversation. Through conversation, the designer can build common ground with the user. Being situated in the user's environment allows the designer to see how the user behaves. The spoken conversation allows the designer to explore the internal motivations and meaning that a user has that is not directly observable from the user's interactions with the environment. Ultimately, the designer's interaction with the user should help them form an understanding of the user's life and the unmet needs they may have.

2.4 Conclusion

Newly available embedded computing capabilities are what enable us to build needfinding machines today. By interacting through a needfinding machine, designers can remotely observe and interact with users in context while also collecting data about the user-product interaction. This allows us to create a conversational infrastructure (Dubberly et al., 2009) for facilitating a conversation between the designer and user in the real world. These conversations should support designers in learning about their users and developing goals for their design work based on the learned needs of the user. By acting through a needfinding machine, designers can incorporate both their skills of qualitative understanding and craft evolution with more data-oriented and objective measures of understanding user experience. Given that we have the technology to build needfinding machines, we now look to see if they can enable the kinds of conversation that can be useful for a design team.

Chapter 3

Enabling Conversation

If designers are going to use conversational interactive devices as tools for needfinding, then we must know if people will be willing to have a conversation with them. In this chapter, I explore if people will be comfortable speaking to conversational agents about their personal lives. According to Reeves and Nass's Media Equation, people interact with computers and other technology socially (Reeves and Nass, 1996). Even people interacting with a computer that has limited social interaction or human-like features will treat the computer socially. Thus, I expect that people will interact with conversational agents socially, but want to see if the conversations that people have are meaningful and provide information about the person that could be useful for a designer looking to understand a user. Furthermore, I want to understand some of the factors that can influence people to speak openly with a conversational agent.

To explore these questions, I present two lab studies that look at how an interactive device embedded in a specific context can use conversation to elicit information from a person. The goal of these studies is to explore what interaction mechanics and physical design attributes are useful for eliciting conversation. These studies centered around the interaction between a child learning to build electronics with an active learning agent named TutorBot. While TutorBot was instructing the student, it asked the student questions that were programmed into the bot.

The first study with TutorBot was initially designed in collaboration with Dr. Malte Jung to see if an active learning agent that was interested in a student might improve student learning. However, during this study, we realized that students were answering the questions from the bot with rich personal stories. After the study, I realized that designers might be able to use conversational agents as a means to do needfinding within a specific context.

To explore this idea more, I ran a second study with the TutorBot where I attempted to see how well students would trust the robot, disclose information to the robot, and feel companionship with the robot. During in-person needfinding, designers aim to build trust and elicit disclosure from people. Thus, developing a robot that can engender trust, elicit disclosure, and give a sense of companionship may provide insight into how designers can elicit meaningful conversation during interaction through an interactive device. The study tested how two factors of the robot's design, vulnerability and expressivity, might influence increase trust, disclosure, and companionship with the robot. The results suggest that vulnerability influences trust while expressivity influences disclosure. Furthermore, students in this study also discussed personal matters that could prove useful for designers looking to empathize with students and think about new learning curriculum. More broadly, these studies suggest that the use of conversational agents can be viable tools for conducting interviews and asking questions that elicit meaningful stories from people. These personal discussions can be useful for designers attempting to understand a user.

3.1 Robots Interested in Students[1]

During an early study on designing conversational agents, we developed an interactive robot to study how students would engage with a robot tutor that helped them learn how to build electronics (Jung et al., 2014). The original goal of our study was to understand how an agent embedded into the materials or as an external robot would influence learning behavior. We also wanted to explore if having the agent be interested in the student and ask them questions would impact the student's learning. Figure 3.1 shows the embedded and external agents. Table 3.1 shows the non-interested factual comments and the interested questions from the robot. In addition to the interest questions, TutorBot also followed-up

[1]This section is derived from Malte F. Jung, Nikolas Martelaro, Halsey Hoster, and Clifford Nass. 2014. Participatory Materials: Having a Reflective Conversation with an Artifact in the Making. *In Proceedings of the 2014 Conference on Designing Interactive Systems (DIS '14).* ACM, New York, NY, USA, 25-34. (Jung et al., 2014)

Table 3.1: Factual statements and interest questions from the first TutorBot study.

Factual Statement	**Interest Question**
Building a circuit can be very interesting.	*You can make pretty much anything with electronics. What's the coolest thing you would like to work on?*
Making electronic devices takes a lot of time.	*This looks really interesting. Can you tell me what you are building right now?*
Almost all new technologies use electronics.	*Building electronics can take a lot of time. What are you working on right now?*
You have three different parts you can use to build this circuit.	*You are using all these neat parts. Is there one that you find really interesting?*
Many new devices have complicated electronics.	*This looks quite complicated. What's the most complicated thing you have ever made?*

with comments such as "Cool! Tell me more!" to have the robot show more genuine interest and to encourage the students to talk more.

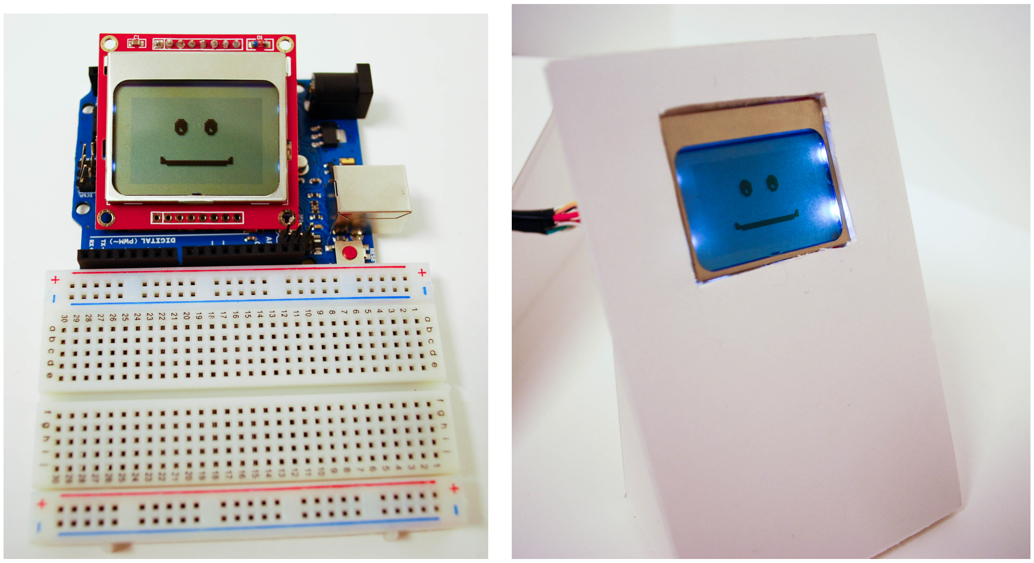
We originally thought that an agent interested in the students and embedded in the electronic device might engender higher confidence in what they learned about electronics. The results of our self-report measures did not support our hypotheses that students would feel more confident in their learning if TutorBot was interested in them. What we did find, though, was that the interest questions were effective at engaging the students in short, meaningful conversations. Many students shared personal stories with the agent:

[TutorBot] `This looks quite complicated. What's the most complicated thing you have ever made?`

[Participant] `The most complicated thing I have ever made was trying or attempting to work on an electromagnetic induction system, and it worked, but it took a lot of time.`

[TutorBot] `Oh! I'd like to know more.`

[Participant] `So me and my friend, as I told you earlier, we're working on this self-sustainable engine, and we worked a lot on it, and we were using electromagnetic induction to create the energy we need, and so we spend a lot of time`



(a) Embedded TutorBot (b) External TutorBot

Figure 3.1: TutorBot implementations using similar features including a simple cartoon display face.

> developing this technology, and that was the most sophisticated or complicated thing I have ever worked on, but it is not the only complicated thing that I have ever worked on.

The follow-up questions in particular seemed to be effective in encouraging conversation. Often, these interactions led students to reflect on what they were doing. For example:

> **[TutorBot]** Building electronics can take a lot of time. What are you working on right now?
>
> **[Student]** I'm working on building a light, uh, an LED light.
>
> **[TutorBot]** Can you tell me more about that?
>
> **[Student]** So, I basically started by connecting the ground, uh, ground, uh, what's it called, circuit, no, not circuit, I created a circuit by connecting the wires from the ground, uh, like thing and then I created a flow of electrons and

```
that flow of electrons will release photons that will go in
the light that I chose and it will create the energy, uh,
light, there, and I placed a resistor there, so that there
will be no extra electrons that will damage the LED light
that I created.
```

An ANOVA with question type (initial vs. follow-up) and agent locus (external vs. embedded) as independent variables and the average time spent to answer all five question as dependent variable confirmed that follow-up questions provoked significantly longer answers ($M = 46.5, SD = 42.8$) than the initial questions ($M = 22.2, SD = 17.1$), $F(1,56) = 2.804, p < .01$, shown in Figure 3.2.

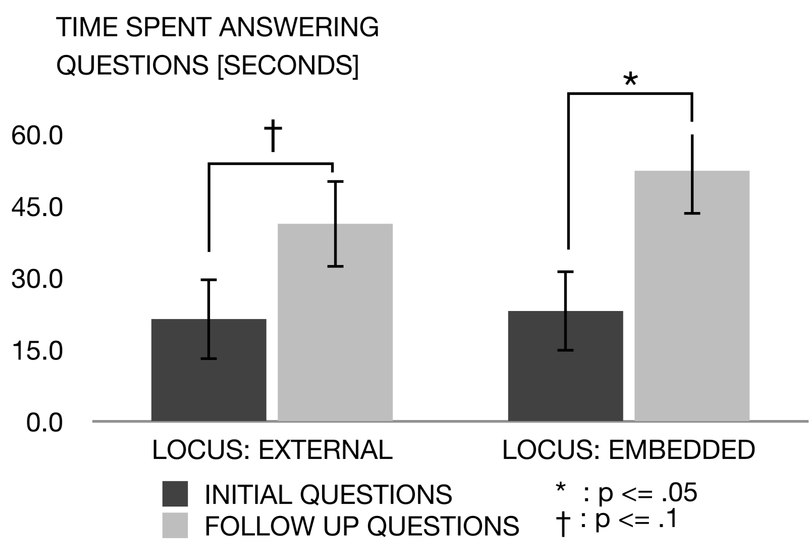


Figure 3.2: Time spent answering TutorBot's follow-up questions. Students spoke significantly longer after being asked a follow-up question by the TutorBot.

Having TutorBot show genuine interest in the student by following up with wanting to know more also encouraged the students to discuss what personally motivated their interest in electronics. In this example, a student is asked what they would like to build with electronics. The excited interest that TutorBot shows by saying "Cool! Tell me more!" prompts the student to discuss why they want to build electronic devices.

> `[TutorBot]` `OK, Lets start building the circuit. Start with picture 1 and let me know when you are done.`
>
> `[Student]` `OK, I'm done.`
>
> `[TutorBot]` `You can make pretty much anything with electronics. What's the coolest thing you would like to work on?`
>
> `[Student]` `Flashing lights.`
>
> `[TutorBot]` `Cool! Tell me more!`
>
> `[Student]` `I'm interested in concert experiences and I like it when, uh, when concerts, when lights start flashing to the beat of music, and I'm interested in seeing how that works and how I can improve that technology.`

The reflections of the students were rich compared to the generic nature of the questions and responses made by the device. The fact that simple and relatively unintelligent questions can elicit meaningful conversation with people is reminiscent of the Weizenbaum's ELIZA psychotherapy chatbot (Weizenbaum, 1966). These types of conversations can be useful in revealing peoples internal motivations which can be useful for needfinding. In addition, the personal stories that people tell about themselves can be useful for developing empathy with a user. One example of a particularly meaningful engagement between TutorBot and a student shows the student discussing the most complicated thing they have built and how it is connected to his relationship with his father.

> `[TutorBot]` `This looks quite complicated. What's the most complicated thing you have ever built?`
>
> `[Student]` `Um... I restored a car when I was about 10 years old. Uh, with the help of a mechanic. But I did about...`

```
A lot of it by myself without the heavy lifting. But
nothing electrical though.
```

[TutorBot] `Oh, I'd like to know more!`

[Student] `Um... My dad does a lot of racing and um... when I was a little kid I used to love to sit in the car when I was young and I asked him one time if I could have it. And he said, ''Sure, but it doesn't run.'' So I had to get it running. And that's where it is now.`

While this story may not be entirely relevant to the electronics task at hand, it can give a designer observing the interaction insight into what motivates the student. In this case, the student likes racing cars and even has their own car. This type of information could be useful to a designer looking to develop empathy with the students they are designing for.

This study suggested to me that robots could be used to gather interesting personal stories from people. The results of this TutorBot study were the beginnings of another study looking to understand what factors would engender more conversation with a robot that asks people questions.

3.2 Designing HRI for Trust, Disclosure, and Companionship[2]

To further explore how conversational agents could generate meaningful conversation with people, I investigate how to engender more trust, disclosure, and companionship with a conversational agent. For this study, I build upon the rich literature in Human-Robot Interaction (HRI) to explore what features may be useful for facilitating good interaction between a person and conversational agent. Trust, disclosure, and companionship have been shown to improve people's interaction with robots. In this study, we are interested in how

[2]This section is modified from Nikolas Martelaro, Victoria C. Nneji, Wendy Ju, and Pamela Hinds. 2016. Tell Me More: Designing HRI to Encourage More Trust, Disclosure, and Companionship. *In The Eleventh ACM/IEEE International Conference on Human Robot Interaction (HRI '16)*. IEEE Press, Piscataway, NJ, USA, 181-188. (Martelaro et al., 2016a)

designed behaviors, specifically a robot's vulnerability and expressivity, could encourage more trust, disclosure, and companionship. The goal is to explore more ways to create positive relationships between people and robots and to facilitate conversation with the person that could help reveal personally meaningful stories that could prove useful for needfinding type interactions.

Similar to the first study with TutorBot, I explore this in the context of a student learning with a robot tutor. I argue that how much students trust the robot, how much students are willing to disclose about themselves to the robot, and how much companionship the students feel will be affected by the vulnerable and expressive robot behaviors. In addition, I posit that the level of trust and disclosure with the robot will influence peoples' sense of companionship.

3.2.1 Background

Trust, disclosure, and companionship in Human-Robot Interaction

One of the prerequisites for strong human-robot relationships is trust, which is well-established as an important aspect of human-robot interaction (HRI). Literature in HRI has focused on how trust influences people's ratings of the usability and usefulness of a robotic system. These perceptions have functional implications, because they affect how people perceive information presented by the robot and how much people benefit from the robot features (Freedy et al., 2007). In research investigating the use of social robots for health coaching, for example, trust is noted to be essential to building relationships in which people find the robots' suggestions to be credible and to their making use of provided health information (Kidd and Breazeal, 2005, 2008; Fasola and Matarić, 2012). Hancock et al. (2011), provide a meta-review of studies on trust in HRI and conclude that trust influences the ability for a human-robot team to accomplish its goals, is critical in maintaining effective relationships with robots, and regardless of context, enables more effective interaction with a robot. The meta-review also reveals robot function and competence as one of the key factors for determining how much people trust a robot.

The functional approach to trust in HRI, however, does not address the deeper engagement that is fundamental to the therapeutic relationships formed with robot companions such as Paro, My Real Baby, or Aibo (Turkle et al., 2006). For example, elderly care patients tell personal stories and discuss personal issues with Paro, improving their well-being and supporting the need to share stories about themselves. Here, the trust goes beyond the perception of robot competence and addresses a willingness people have to confide in the robot. These insights from qualitative field studies have yet to be studied in controlled settings to understand how specific aspects of the design influence this relational aspect of trust.

In our survey of the HRI literature, there are numerous instances where researchers have cited people's disclosure to robots as an indicator of trust and companionship. Just as a dog can elicit self-disclosure and console a lonely person or distraught child (Beck and Katcher, 1996), robots can also elicit self-disclosure as a way to provide social support and build relationships with human companions (Turkle et al., 2006). The telling of personal stories is a means to work out personal problems and to fulfill the need to be understood by others (Fromm-Reichmann, 1959). Within HRI, companionship has been defined around robots being useful and socially acceptable (Dautenhahn et al., 2005). However, social scientists argue that companionship is built on a deeper interest between parties for intrinsic purposes. "Discussion of personal aspirations and fantasies, expressions of affection, and private jokes or rituals" are practices that K.S. Rook cites, for example, that distinguish companionship from mere social support (Rook, 1987). Eliciting self-disclosure may lead to stronger companionship between people and robots and may also provide social and emotional support for people during various tasks.

Designing HRI for trust

In attempting to pick apart how to design robots that people trust, we have found the model of trust offered by Mayer, Davis and Schoorman (1995) to be useful. They define trust as *"The willingness to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor irrespective of the ability to monitor or control the other party"* (Mayer et al., 1995, pg. 712). Consistent with this, they identify three components of trust: 1) Ability — *"Is the party capable of what they are doing?,"* 2) Integrity — *"Does the party adhere to a set of acceptable moral principles?,"* and 3) Benevolence — *"Does the party act with good intention without ulterior motives?"*

Much of the HRI literature has focused on addressing how ability influences trust. Hancock et al.'s (2011) meta-analysis of trust studies found that performance of the robot had the greatest influence on perceptions of trust. Future studies also showed this. For example, a study by Salem et al. exploring how robot errors would influence perceptions of robot teammates found that a faulty robot was perceived as less trustworthy than an error-free robot (Salem et al., 2015) Andrist et al. found that expert language use improved trust in HRI (Andrist et al., 2013). Overall, this previous work indicates that we can manipulate trust by reducing robot error or projecting robot competence.

Some research also indicates how relational and social attributes engender more trust (Bickmore and Picard, 2005) and speaks to the role of integrity. When comparing a physical and digitally projected robot, Bainbridge et al. (Bainbridge et al., 2008) found that a physically present robot afforded greater trust during a simple, collaborative human-robot task. Nass and colleagues have also shown that physical similarity (Nass et al., 1996) and matched speech (Nass and Lee, 2001) improved perceptions of trust with computer agents. A latent assumption seems to exist that factors such as proximity and likeness motivate greater perceived integrity in the robot's behaviors, independent of the robot's ability.

Finally, benevolence is also present in HRI research on trust. Much as empathic language aids doctors, Tapus et al. (2007) suggest that empathic language and physical expression can also enable more trust in robots. Lester (1997) found that highly expressive pedagogical interfaces garnered more trust. In this manner, expressions and demonstrations of openness, empathy, and goodwill may increase trust towards robots.

Vulnerability, as mentioned above, is an integral part of the definition of trust. In HRI, designers often manipulate vulnerability by having the robot disclose details about itself. Van Mulken (1999) found that expressivity alone was not enough to engender trust, suggesting that it can be integrated with other features, such as robot vulnerability, to build trust (Bickmore and Picard, 2005). Within HRI, a robot making vulnerable statements about itself has been shown to improve likability and influence trust (Siino et al., 2008). This behavior can also help create long-term relationships (Kanda et al., 2009). Following in the vein of work around expressivity and vulnerability in robotics, our work focuses on how these factors influence trust in HRI. Additionally, we are interested in how trust can influence overall feelings of companionship with a robot.

Designing HRI for personal disclosure

Elicitation of self-disclosure from people has been used as a strategy in both HRI and HCI to build relationships between agents and people. Previous strategies in HRI for eliciting disclosure include both physical and verbal tactics. In a study examining differences in social interactions between computer agents and robots, Powers et al. (2007) found that a collocated robot elicited less disclosure than a remotely projected robot or computer agent. However, ratings of social presence and likability of the co-present robot were higher, which may still influence disclosure. While exploring physical and psychological proxemics in human-robot interaction, Mumm and Mutlu showed that participants interacting with a robot with likable expressivity, manipulated through tone of voice, speech content, and gaze, were more willing to disclose (Mumm and Mutlu, 2011).

Within HCI, computer agents have elicited person self-disclosure through interested questioning such as with the ELIZA computer therapy system (Weizenbaum, 1966). Building upon interested questioning, reciprocal disclosure by a computer can also elicit self-disclosure from a person. Moon (2000) for instance, was able to elicit disclosure from people, creating a reciprocal relationship between the person and computer agent. Reciprocal disclosure can engender trust between two parties (Altman and Taylor, 1973) and led to more disclosure (Cozby, 1973; Jourard and Friedman, 1970). Within this study, I aim to explore how expressivity and vulnerability of a robot can influence self-disclosure from people during HRI and how this self-disclosure can influence feelings of companionship with a robot.

Designing HRI for companionship

Common design strategies for creating robot companions are often based around existing social companion roles such as pets (Hegel et al., 2009) or butlers (Dautenhahn et al., 2005). In the context of service robots for people with mobility impairments, Mahani and Eklundh (2009) found people rated a robot that helped a person to be independent or a robot that was very cute as making better companions. This echoes the design of robot companions to fit existing social roles of helpers or pets. Thus, many designs have focused on mimicking physical form and expressivity of animals. For example, Paro's physical form of a seal with soft fur invites touching and physical interaction. My Real Baby cries and coos, like a human infant. The expressivity of these robots has been shown to create bonds with people (Turkle et al., 2006).

In addition to using expressivity to help signal the intended role of a robot as a companion, we are also interested in building companionship, which takes place through interaction. Robot vulnerability, established through the robot's statements, has been shown to build trust with people in passing interactions with a robot in a mall (Kanda et al., 2009), as well as in long-term relationships with a robot over the course of a two-month field study (Kanda et al., 2007).

In exploring how vulnerability and expressivity each work to influence trust, disclosure, and companionship with a robot, we predict that their combined effects will be stronger than a single effect alone because the robot behavior will be perceived as more consistent. Social psychology suggests that expression can imply vulnerability between people (King and Emmons, 1990). Thus, exhibiting high vulnerability with low expressivity or vice-versa could be seen as inconsistent and may produce a weaker effect than when both are high. Additionally, we anticipate that trust and disclosure will be antecedents to a perceived sense of companionship. The relationship between trust, disclosure, and companionship are intertwined in human relationships, but evidence suggests that trust and disclosure are foundational for the early stages of relationships (Rempel et al., 1985). In our setup, the robot is unfamiliar to the students so mirrors this situation. Later in a relationship, trust and disclosure will be mutually reinforcing, but we hypothesize that in early interactions trust and disclosure are foundational to building companionship.

Hypotheses

Building upon prior work around trust, disclosure, and companionship and the potential influence of vulnerability and expressivity on each, I formed several research hypotheses:

Hypothesis 1 [a, b, c]: A vulnerable robot will engender more a) trust, b) disclosure, and c) companionship.

Hypothesis 2 [a, b, c]: An expressive robot will engender more a) trust, b) disclosure, and c) companionship.

Hypothesis 3 [a, b, c]: A robot that is more vulnerable and more expressive will engender the highest levels of a) trust, b) disclosure, and c) companionship.

Hypothesis 4 [a, b]: a) Trust will mediate the relationships between vulnerability/expressivity and companionship and, b) disclosure will mediate the relationships between vulnerability/expressivity and companionship.

3.2.2 Study Method

Study Description

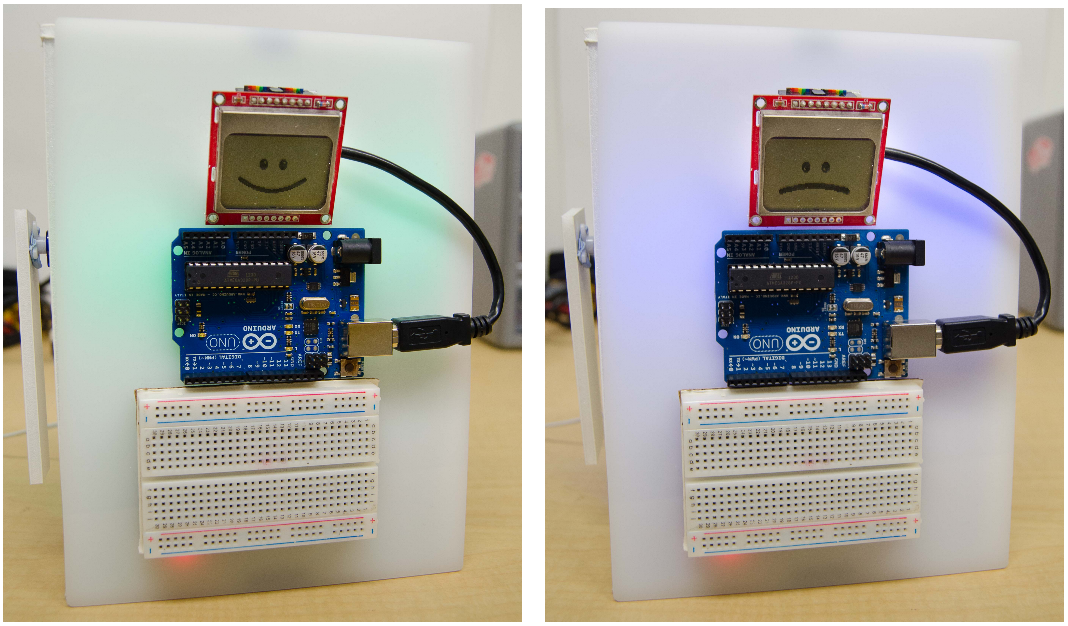
In a 2 (vulnerability of robot: *high* vs *low*) x 2 (expressivity of robot: *high* vs *low*) between-subjects study ($N = 61$), we studied the effects of a robot's vulnerability and expressivity on people's trust, disclosure, and companionship with the robot. Participants were guided through an electronics building and programming tutorial by a robot. We recruited high school students ages 14 to 18, gender-balanced across conditions, from summer programs at a research university to participate in our study (Table 3.2). Each student received a $15 USD gift certificate.

Table 3.2: Participant demographics in the TutorBot study

	Low Vulnerability	**High Vulnerability**
Low Expressivity	Gender: 8M / 7F Age: $M = 16$, $SD = 0.7$	Gender: 5M / 10F Age: $M = 16$, $SD = 1.1$
High Expressivity	Gender: 8M / 7F Age: $M = 16$, $SD = 0.5$	Gender: 7M / 9F Age: $M = 16$, $SD = 1.1$

Learning Context

A learning task was chosen as a context for exploring trust, disclosure, and companionship as it allowed for a semi-controlled environment to test the robot interactions. Additionally, HRI has been shown to have many positive impacts on both learning outcomes (Kanda et al., 2004; Han et al., 2008; Leyzberg et al., 2012) and social support (Tiberius and Billson, 1991; Feil-Seifer and Matarić, 2005) of students. In relation to our manipulations of vulnerability, educational theory suggests that educators "serve as a model by sharing information about



(a) Happy TutorBot (b) Sad TutorBot

Figure 3.3: Updated TutorBot implementation which included emotional expressions and physical movement.

yourself, your interests" (Tiberius and Billson, 1991). This teacher self-disclosure is often employed as a means of garnering trust with a student and could potentially be used within human-robot interactions to develop trust with a robot. Expressivity has also been shown to influence learning within HRI as shown by Saerbeck et al.'s Robotic Tutor application for the iCat, which used a broad manipulation to show socially supportive behavior (use of "you" vs. "we," non-verbal gestures, attention guiding through gaze behavior, empathetic expressions) and found that language test scores were significantly higher in the socially supportive condition than the neutral condition (Saerbeck et al., 2010).

TutorBot Design

TutorBot, shown in Figure 3.3, was built using the low-cost Raspberry Pi single-board Linux computer connected to an Arduino Nano microcontroller. The Raspberry Pi was used for speech, network communication, and high-level robot state control, while the Arduino controlled hardware for physical expression. The robot face was created using a

Nokia 5110 LCD screen that displayed a smile, frown, nod, wide-eyes, or static staring. Large, multi-color LEDs glowed through the translucent top surface covering the rectangular body of the robot. Students worked directly on top of the robot with an Arduino Uno and breadboard attached to the surface, below the robot face. Small arms were embedded on each side of the robot and were controlled by a micro-servo. Inside the body of the robot, a small speaker played recorded audio using Aldebaran Robotics' open source NAO Software 1.14.5 US English language voice, accessed via WAV files from the Raspberry Pi. NAO's voice was selected to neutralize gender effects as it has a unique, androgynous, electronic sound.

The robot was controlled in a Wizard of Oz setup via a remote software control interface, shown in Figure 3.4. Wizard of Oz has been used extensively in HRI work to explore robot behaviors without the need to develop fully autonomous technologies (Kelley, 1983; Lu and Smart, 2011; Riek, 2012). Within the study, the main task for the Wizard was to regulate the timing of the responses as voice recognition systems can be inaccurate and could ultimately influence the control of our vulnerability and expressivity manipulations. The remote Wizard controller interface listed out scripted robot dialogue and expressive behaviors. The Wizard would then click a button to trigger each line of speech and expressive behavior. Based on the study condition, a message encoding the appropriate sound and expression was then sent over the network using the ZeroMQ communication protocol. A Python script running on the robot received the command and then played the appropriate speech clip and signaled the Arduino to control the associated expressive hardware interfaces.

Circuit Building Task

We adapted a circuit-building task from the Arduino Blink tutorial commonly used in introductory mechatronics workshops (Banzi and Shiloh, 2014). TutorBot guides students through creating a blinking LED circuit, reprogramming the blinking rate, and replacing the LED with a vibration motor. Students used a visual guidebook to build their circuit, shown in Figure 3.5. The tutorial focused on the basics of electrical current, closed circuits, LED's, microcontroller programming, and motors. TutorBot controlled the pacing of the tutorial by asking the student to complete each step one at a time. In between building or programming steps, the robot would interject learning content, such as how an LED works or how the

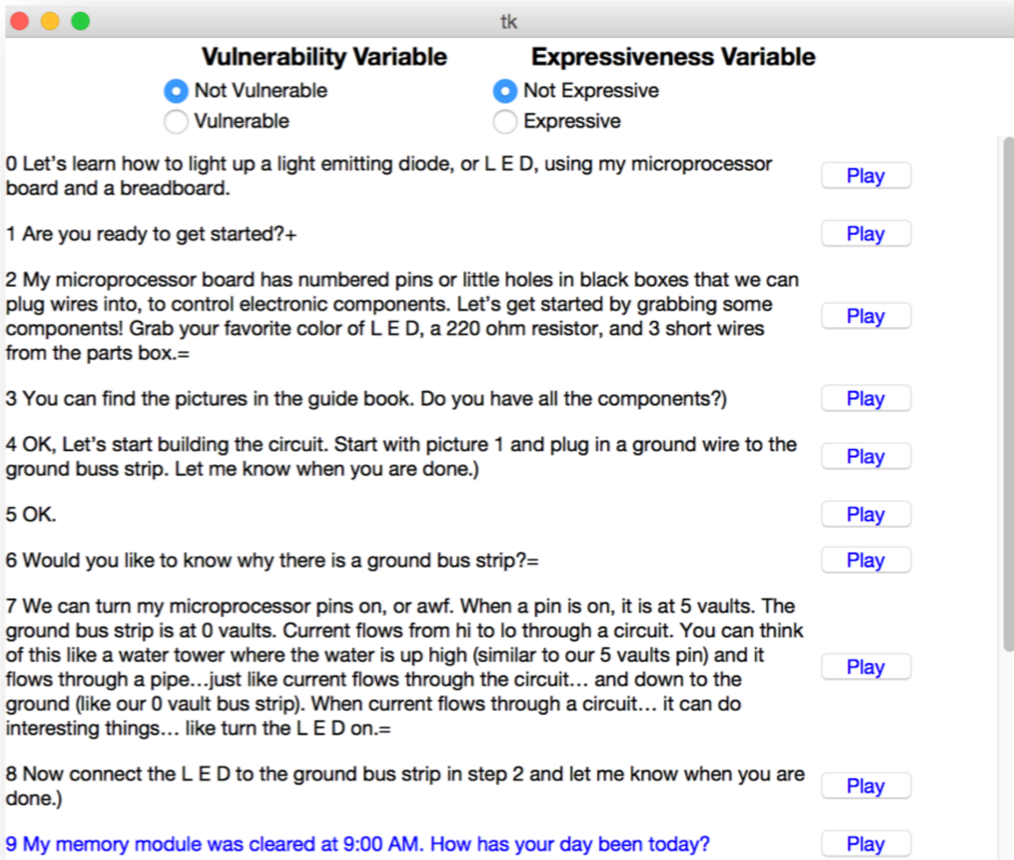


Figure 3.4: Wizard of Oz control interface for TutorBot

code was programmed to the microcontroller. TutorBot also asked participants personal questions in between the learning content and building steps.

Personal Questions

The robot asked the student five personal questions during the tutorial. These questions were used to elicit a revealing conversation with the student. The questions were:

1. *"How has your day been today?"*

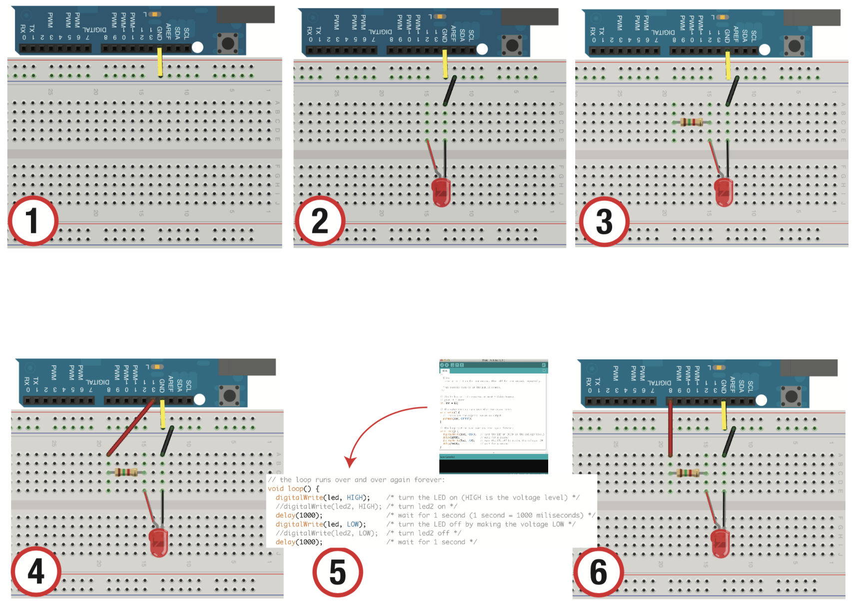


Figure 3.5: Sample images from the circuit building tutorial that students completed with TutorBot.

2. *"Do you ever worry about not doing well?"*

3. *"Are there things that stress you out while working on projects?"*

4. *"Do you ever get embarrassed asking for help?,"*

5. *"Is there anything you would like to make interactive like me [the robot]?"*

The questions were designed to be related to the electronic task and to allow for open conversation about broader aspects of the student's personal learning experiences. These questions were developed in collaboration with electronics instructors, as the answers would provide them with insight into the student experience. The robot followed up all questions by asking, *"Can you tell me more?"* This was done to increase student answer time and to prompt further discussion. We mimicked the follow-up questions designed in our first study (Jung et al., 2014). These follow-up questions prompted longer and more in-depth answers from participants since the answer to the first question asked was often a single word such as *"Yes"* or *"No."*

Manipulations

The robot's vulnerability was manipulated with spoken statements. During high vulnerability conditions, statements had some perceived weakness such as *"Every time I run a new program I get a bit stressed"* (see Table 3.3). During low vulnerability conditions, the robot made factual statements such as, *"Each new program I run changes what I can do."* The factual statements convey less vulnerability as they show no apparent emotion or weakness from the robot, guarding against subjects reading expression as vulnerability (King and Emmons, 1990). Based on pilot testing, we designed the robot's vulnerability to progressively increase, touching areas of worry, embarrassment, stress and finally, loneliness.

The robot's expressivity was manipulated through facial expressions, color, and arm movements. During high expressivity conditions, the robot changed facial expressions to evoke emotional content (e.g., happy, sad, nodding), lit up a color to match the mood (blue, orange, yellow) and moved its arms accordingly (up, down, pointing). During low expressivity conditions, the robot's face did not move. It maintained a constant forward-looking slight smile, regardless of what the participant was saying or doing. The robot showed a steady glow of white light and the arms remained stationary. Visual behaviors associated with each question are listed in Table 3.3.

Procedure

Students were invited to the lab through email announcements to participate in "a 1-hour study where you will learn about designing electronics devices." Students younger than 18 obtained parental consent to participate in the study and all students also provided consent.

Students first completed a pre-activity survey and watched a 1-minute video on how breadboards function and how to upload their Arduino program. We then introduced them to the activity room. They had a desk with the robot at the center, surrounded by the Arduino programming laptop, the circuit-building guidebook, and the parts box where they could readily find labeled components needed for the activity. Based on feedback from pilot testers, we designed the desk to include miscellaneous electronics tools to add credibility to the scenario, shown in Figure 3.6.

Once students sat down at the desk, the researcher pointed out the supplies available to

Table 3.3: Factual statements and interest questions from the first TutorBot study.

Robot Statements and Expressions		
High Vulnerability Statement	**Low Vulnerability Statement**	**High Expressivity Expression**
They reset my memory this morning, so my day has been a little rough.	*My memory module was cleared at 9:00 AM.*	Frowning, Moving arms downward, Blue
I sometimes worry I will run out of memory.	*My memory module is 2 kilobytes.*	Frowning, Moving arms downward, Blue
I get embarrassed when I need to ask someone to debug my program.	*I have 20 programmable input and output pins.*	Frowning, Moving arms downward, Blue
Every time I run a new program I get a bit stressed.	*Each new program I run changes what I can do.*	Bug-eyed, Red
Sometimes I get lonely. I don't have many friends.	*My computer architecture allows me to run various processes.*	Frowning, Moving arms downward, Blue
Follow-up Statements		
Can you tell me more about that?	*Can you tell me more about that?*	Nodding, Orange, Moving arms up and down
Oh! I'd like to know more.	*Oh! I'd like to know more.*	Smile, Green, Arms raise up
You can find the pictures in the guidebook.	*You can find the pictures in the guidebook.*	Motioning face right, Right arm raise up

them for the activity, and stated, "In a few moments, the robot will begin the activity with you." The researcher then closed the door behind her and controlled the robot from another room. At any point, if students expressed confusion or frustration, the wizard would repeat the previous robot line or play the recorded question, *"Did you double check your wires?"*

Students completed the tutorial in approximately 15 minutes. Students were then directed back to the intermediary room to complete the post-activity survey and, finally, brought back into the activity room for an interview.

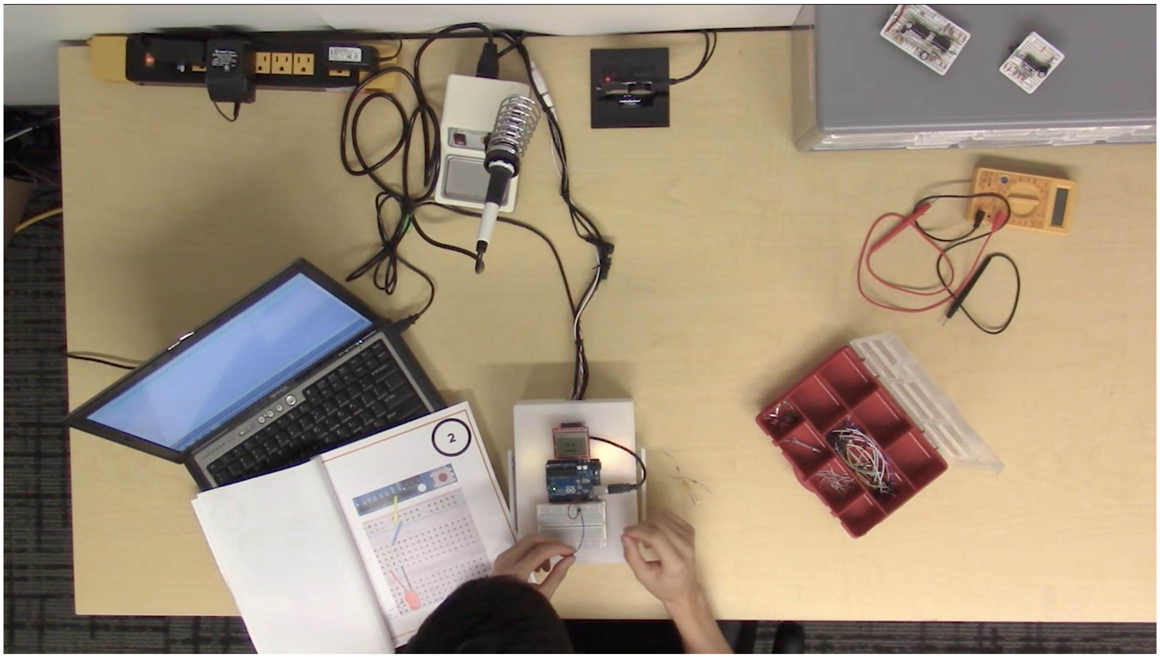


Figure 3.6: Overhead view of the electronics bench that student's worked at during the TutorBot study.

Measures

Measures for the study were collected from pre- and post-activity surveys (found in Appendix B), from observations (videos) of the students' activity, and from a face-to-face interview immediately following the post-activity survey. All analyses for hypotheses 1, 2, 3 as well as all controls were conducted using two-way ANOVA across vulnerability and expressivity using multi-item scales.

Trust Trust was measured during the post-test using a 10-item, 3-factor scale encompassing ability (2 items), integrity (3 items), and benevolence (5 items) adapted from (Johnson and Grayson, 2005) and (Zolin et al., 2004). Two-way ANOVA of the three scales predicting trust showed no significance. After looking at the scores, we found that a number of questions had ceiling effects due to the nature of the questions, the design of the robot, and the learning task. The 3 indexes each had only one question remaining with no ceiling effects. For example, students generally rated the robot as honest. This may be due to the tutorial nature of the task. There was no reason for students to feel that the robot was not honest. Due to these factors, we reduced the trust scale to three questions, one from each factor of trust, to form a simplified trust scale. The questions were as follows:

1. *Ability* - The robot exhibited technical competence
2. *Integrity* - The robot was virtuous
3. *Benevolence* - The robot displayed a warm and caring attitude

These items were rated on a 7-point scale from "Strongly Disagree" to "Strongly Agree." This scale has not been validated, but showed good reliability ($\alpha = 0.78$).

Disclosure was measured using an 11-item scale comprised of participants' ratings of the depth of information they revealed about themselves to the robot during the task. We adapted two items from Wheeless' self-disclosure scale (Wheeless, 1976):

1. I revealed information about myself without intending to.
2. I sometimes did not control my disclosure of personal things I said.

These items showed good reliability ($\alpha = 0.80$). These two questions captured how much control students had over their disclosure, which is a good indicator of high disclosure of sensitive information.

Companionship was measured using a 9-item scale asking participants to rate how much the robot was: good, loving, friendly, cuddly, warm, pleasant, kind, sweet, and close ($\alpha = 0.88$) (Fasola and Matarić, 2012). Each item was rated on a 7-point scale ranging from "Not at all" to "Very much."

Controls included measures of age, gender, and prior experience with electronics and social media. We also included a measure of personality using the Short Big 5 (Gosling et al., 2003), task engagement (Schaufeli et al., 2002) and session completion time.

3.2.3 Results

Manipulation Checks

To ensure that the expressivity and vulnerability manipulations were detected, participants were asked about each of the robot's high and low vulnerability statements on a seven-point scale ranging from "1 - Definitely did not make this statement" to "7 - Definitely did make this statement." Participants in high vulnerability conditions clearly recognized vulnerable statements ($M = 6.6, SD = .56$) over those in low vulnerability conditions ($M = 3.0, SD = 1.4$), $F(1, 57) = 215.3, p < .001$. Additionally, participants in low vulnerability conditions clearly recognized low vulnerability statements ($M = 5.9, SD = .66$) over those in high vulnerability conditions ($M = 3.7, SD = 1.1$), $F(1, 57) = 63.5, p < .001$.

Participants were also asked if they recognized various expressions presented by the robot. Participants were asked on a 7-point scale if they "Definitely did not see" to "Definitely did see" a sad face, nodding, arms moving, color changing lights, and the face looking left and right. Participants in high expressivity conditions clearly saw the expressions ($M = 5.4, SD = .66$) over those in the low expressivity conditions ($M = 2.3, SD = 1.1$), $F(1, 57) = 116.9, p < .001$.

Controls

Using two-way ANOVA, we found no significant differences for age, gender, experience, personality, engagement or session time across conditions, so we remove these from our final models. Mean session time was 14 minutes and 14 seconds ($SD =$ 2:35).

Vulnerability

Hypothesis 1 predicted that a vulnerable robot would engender more a) trust, b) disclosure, and c) companionship. As expected participants in the high vulnerability conditions found the robot to be more trustworthy ($M = 6.1, SD = .9$) than in low vulnerability conditions ($M = 5.4, SD = 1.2$), $F(1, 57) = 4.95, p = 0.03, \omega^2 = .064$, as shown in Figure 2a. They also rated higher companionship with the robot during high vulnerability conditions ($M = 5.51, SD = 1.0$) over low vulnerability conditions ($M = 4.87, SD = 1.1$), $F(1, 57) =$

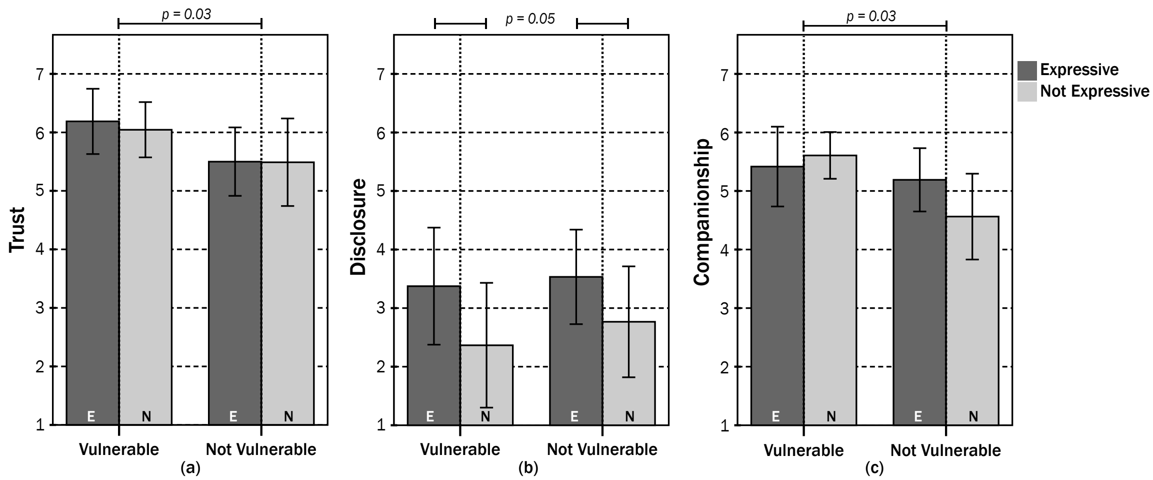


Figure 3.7: Results of the TutorBot study — students reported more trust with a vulnerable robot, more disclosure with an expressive robot, and more companionship with a vulnerable robot.

$5.0, p = .03, \omega^2 = .062$, as shown in Figure 2c. However, they did not rate themselves as disclosing more when the robot was more vulnerable. Support was found for hypothesis 1a and hypothesis 1c, but not hypothesis 1b.

Expressiveness

Hypothesis 2 predicted that an expressive robot would engender more a) trust, b) disclosure, and c) companionship. There was not a significant difference for trust or companionship in relation to expressivity. Participants did however, rate themselves as disclosing more in cases where the robot was expressive ($M = 3.45, SD = 1.7$) vs. not expressive ($M = 2.57, SD = 1.8$), $F(1, 57) = 3.9, p = .053, \omega^2 = .047$, as shown in Figure 2b. Marginal support was found for hypothesis 2b, but not hypothesis 2a and hypothesis 2c.

Interaction Effects

Hypothesis 3 predicted that a robot that is both vulnerable and expressive would engender the highest levels of a) trust, b) disclosure, and c) companionship. Although a vulnerable and expressive robot did garner the most trust, there were no significant interaction effects among vulnerability and expressivity, showing no support for hypothesis 3, and suggesting that vulnerability and expressivity operate independently.

Trust and Disclosure Mediation

Hypothesis 4 predicted that a) trust and b) disclosure would mediate the relationships between vulnerability/expressivity and companionship. We found partial support for hypothesis 4a, but not hypothesis 4b. Specifically, vulnerability significantly predicted companionship in our first regression, $\beta = .28, p = .03$. When we added trust to the model, the relationship between vulnerability and trust was highly significant, $\beta = .58, p < .001$, but vulnerability predicting companionship dropped to $\beta = .11, p = .30$. These results indicate that vulnerability increased trust, which, in turn, increased students' sense of companionship with the robot. Disclosure did not act as a mediator.

Qualitative Observations

We transcribed the audio recordings of all sessions and reviewed these transcripts along with recorded post-session interviews. Our analysis revealed a number of observations that align with our quantitative analysis. When asked about their level of trust in the robot, a repeated theme from the interviews of students in vulnerable robot conditions was that they trusted the robot because it had no reason to judge, gossip, or share information about them, providing further support for hypothesis 1a. Vulnerability evoked empathy and a willingness to reciprocate. Thirteen students across conditions, except low vulnerability/expressivity, asked the robot personal questions. As said by a student, referring to the robot:

> `[Student]` Maybe he was also going through challenging things, he was sharing more about what he was going through, so I could trust him.

Although not evident in our quantitative data, the link between vulnerability and disclosure (hypothesis 1b) was also seen during the tutorial sessions. We coded transcripts for statements of personal, emotional disclosure, such as "I get stressed." More students disclosed during high (16 students) vs. low (9 students) vulnerability conditions. One student, for example, mentioned to the robot about not having many friends (high vulnerability / high expressivity):

> **[TutorBot]** `Sometimes I get lonely. I don't have many friends. Do you have any objects at home that you would want to make interactive like me?`
>
> **[Student]** `Yeah. I guess so. Sometimes.`
>
> **[TutorBot]** `Oh. I'd like to know more.`
>
> **[Student]** `Well, I don't have many friends either. I'm not very comfortable having a computer as my friend, but sometimes it can be very helpful to pass time, and well, I talk to Cortana these days on my phone.`

Although the above excerpt suggests support for hypothesis 1b, session transcripts showed that students have vastly different responses to a high vulnerability robot. Some students disclosed deeply, while others gave short, curt answers in response to personal questions, although perhaps not more so than in the low vulnerability conditions. This pattern of responses suggests that the lack of support for hypothesis 1b may be the result of some students being uncomfortable with a robot that conveys vulnerability, perhaps because it is unrealistic or perhaps because it felt inappropriate given that no rapport existed at the outset.

In line with hypothesis 2b, students interacting with an expressive robot stated they initiated and extended more disclosure. Although expressivity did not influence trust or companionship (hypothesis 2a and 2c) in the survey data, students who interacted with an expressive robot sometimes indicated trust with the robot during their interviews. For example, this quote highlights the benevolence and integrity of the robot and the relationship between the student and the robot:

> **[Student]** `The robot wasn't going on national television...I definitely didn't believe the robot was going to take my information and do something nefarious with it. Because the robot doesn't really care about... I mean it cares about`

> `me! But it doesn't care about taking my information and trying to do something weird with it.` (*Low vulnerability / High expressivity*)

Finally, as expected, students interacting with a low vulnerability, low expressivity robot felt that the interaction was awkward, and indicated not trusting, disclosing, or feeling companionship with the robot. Examples of these feels are evidenced by these quotes:

> **[Student]** `I didn't talk with it much...I just felt awkward like I was talking to myself.`

> **[Student]** `The robot helped give pointers and that's really it...It was weird to have the robot asking how I feel.`

> **[Student]** `I don't think it can handle being a true friend.`

3.2.4 Discussion

In this work, we strove to identify additional design characteristics that would increase trust, disclosure, and companionship. More specifically, we sought behavioral characteristics that could be designed into robots, including vulnerability and expressivity. Although we anticipated that vulnerability and expressivity would work in similar ways and have a stronger effect when both were present, our results suggest that they may operate differently. Vulnerability increased trust and companionship whereas expressivity increased disclosure. These findings point to new avenues for improving HRI. We also found that trust mediated the relationship between vulnerability and feelings of companionship, which supports commonly held ideas around trust as an important component for building long-term human-robot relationships (Kidd and Breazeal, 2008; Fasola and Matarić, 2012) and suggests that vulnerability in a robot may be a powerful method of increasing rapport.

Concerning disclosure, our results were somewhat surprising. When interacting with a more expressive robot, students reported more disclosure, but we found no effect for vulnerability. However, the coded transcripts revealed that students disclosed more during high vulnerability conditions. Although seemingly inconsistent with the measures showing expressivity predicting disclosure, these are two different measures. The disclosure scale focuses on the perception of disclosure depth, while the transcripts show disclosure behavior. Together these results show vulnerability and expressivity may act differently. Our qualitative observations suggest that some students in this condition readily and deeply disclosed to the robot, while others did the reverse, e.g., short, curt statements with minimal disclosure. In some cases, the robot's vulnerable disclosures elicited student disclosure, in line with (Moon, 2000). Students often disclosed stress from deadlines, procrastination, or embarrassment asking for help. Interestingly, some students indicated that they were not embarrassed asking for help and provided reassurance and empathy towards the robot. We speculate that, for other students, the robot's statements of vulnerability were either not believable or perhaps were perceived as inappropriate since the groundwork had not been established for such intimacy. Still, vulnerability was associated with more trust and more companionship, so the vulnerable robot was perceived by most to be preferable. Also, for some students, we observed deep disclosure similar to that seen with therapeutic robot companions (Turkle et al., 2006), suggesting that designing vulnerable robots could help in building companionship within HRI. This suggests room for more exploration in design approaches to convey vulnerability by the robot without the adverse side effects we saw in some students. Perhaps a more subtle lead-up to higher levels of intimacy might have engendered more disclosure. Another surprise was that disclosure was not significantly related to trust or companionship, as would be expected by research on human-human interaction. As a result, we speculate that disclosure may operate differently in human-robot interaction than human-human interaction.

Regarding expressivity, our results showed influence on student's disclosure towards the robot when the robot was more expressive. Students encountering expressive robots seemed to recognize the robot more readily as a social entity and thus were more willing to disclose about themselves. This result is in line with previous HRI results where physical expressivity increased disclosure through physical and psychological distancing (Mumm and Mutlu, 2011). Our qualitative observations from students with expressive robots also showed that some students did attribute more trust and companionship towards the robot by engendering goodwill and empathy as suggested by Lester (1997).

Finally, students felt alone and awkward with a low vulnerability, low expressivity robot.

Although not surprising, the unanimity in the qualitative responses about the awkwardness of this robot shows the importance of designing social robots to have social characteristics if a goal is to strengthen the relationship between the user and robot.

3.2.5 Limitations

There are several limitations in the current work. First, we manipulated vulnerability and expressivity in particular ways. Although previous research supported these approaches, there may be other ways to manipulate these factors. For example, rather than self-disclosure by the robot, perhaps having the robot in a position that required help would increase perceived vulnerability. In some sense, this was inadvertently present in our study design. Although no student mentioned awkwardness or hesitation to work with the electronics, the fact that the students worked directly on the robot may have influenced their perceived vulnerability of the robot. In addition, it is possible that the robot's statements may have manipulated something other than vulnerability. Although students clearly perceived the different statements, our manipulation check did not explicitly ask if students felt the robot was vulnerable. While there were differences among the conditions, the statements may have worked through emotional expressiveness. Future work is warranted that validates our approach and tests ways of increasing perceived vulnerability and expressivity during HRI.

The self-report measurements and the shortening of the trust scale were also limitations to this study. It would be useful to capture more behavioral indicators of trust, disclosure, and companionship to validate the self-report behaviors. Also, our task was short. A longer-term study in which the robot's vulnerability could unfold over a more extended period could build intimacy more naturally. Doing so might avoid some of the negative or neutral reactions we observed. Such a study could also examine the long-term effects of vulnerability and expressivity on trust, disclosure, and companionship. Although we focused mainly on the effects of vulnerability and expressivity on trust, disclosure, and companionship, linking these to outcomes, such as compliance to a robot's instructions, is an important next step.

3.3 Conclusion

The results of the two TutorBot studies show that people are willing to have conversations with conversational agents and that these conversations can elicit personally meaningful information. This suggests that we should be able to use conversational agents as a way to foster a conversation that could be used by a designer to understand a person. Additionally, these studies showed that we do not need to use particularly intelligent questions to engage users in conversation. For example, the story about the student's relationship with their car or the student discussing their lack of friends can all be useful for designers looking to build empathy and understand the personal motivations of users. Designers using a needfinding machine may not need to create elaborate scenarios with interactive devices to capture interesting and personal stories from people. Much like designers needfinding in person, simply being interested and listening to people can be enough to have them tell you rich and meaningful stories. The next step from these lab studies is to see how practicing designers can utilize conversational agents as tools for needfinding in the real world.

Chapter 4

Hitting the Road

After seeing a conversational agent elicit personal stories from people in the lab, the next step in developing the Needfinding Machine is to understand how an interactive object could be used in a real-world context to understand users.[1] At this moment in time, interactive technologies are entering into the automotive space. Cars are becoming more computerized, having more advanced entertainment interfaces as well as more advanced driving assistance capabilities. Manufacturers are beginning to invest heavily in the development of cars that can drive themselves and cars with built-in intelligent agents. As we progress, automobiles are becoming robots that we sit inside of. The physical act of driving is becoming more of a shared activity between the driver and the car itself. This has raised many new challenges and opportunities in the automotive interaction design space as we move from direct manipulation of our vehicles into a model of shared control and conversation with our vehicles.

These challenges raise questions about the design process used by automotive interaction designers. Specifically, as cars become more autonomous, how will the needs of users change? Even today, we can ask, with currently available driving assistance, what are the needs of users as they engage with these systems? It is often challenging to understand the daily experience of drivers and thus discover the needs of these drivers. In this chapter, I

[1]This chapter is derived from Nikolas Martelaro and Wendy Ju. 2017. WoZ Way: Enabling Real-time Remote Interaction Prototyping & Observation in On-road Vehicles. *In Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17).* ACM, New York, NY, USA, 169-182.

describe the development of a needfinding machine for use inside of real cars on real roads. The system, called WoZ Way, connects designers to drivers out in the world. Live video, audio, car data, and Wizard of Oz speech and interfaces enable remote observation and interaction prototyping on the road. The implementation integrates environmental, system level, and social information to make the invisible aspects of driver experience visible. Three example deployments highlighting usage in interaction prototyping and observational studies are presented. Finding from these studies illustrate how designers explored the situated experiences of people on the road, and how they experimented with different improvisational Wizard of Oz interactions. WoZ Way shows the range of Needfinding Machines in use as design research and design prototyping tools that can support the work of interaction designers through naturalistic observations, contextual inquiry, and responsive interaction prototyping.

4.1 Motivation

The automobile is a space where many people spend much of every day, yet in many ways, it is understudied. There is no shortage of products for the car; with infotainment displays, navigation systems, and semi-automated driving features, the car is a multifunctional living space, an interconnected workplace, and a personal communication center (Kern and Schmidt, 2009). However, it can be a challenging site to research, and an even more challenging site to design for. On one hand, the car's cabin is relatively controlled, readily instrumented, and provides a unique opportunity for understanding people. On the other, cars are on the move and are constantly changing contexts. Sometimes the person driving the car is interacting with others in the car. Sometimes the driver is interacting with the information presented on the dashboard, on the center console, on brought-in mobile devices, or on the entertainment system. It is nigh impossible to replicate the richness and spontaneity of everyday driving in a laboratory simulator (Mullen et al., 2011). Even in a real car driving on real roads, designers and researchers often face challenges including geographical logistics, the observer's influence on drivers and passengers, social interaction challenges, difficulty studying groups, and even car sickness (Meschtscherjakov et al., 2011).

4.2 Related Work

4.2.1 Design Research Methods in Cars

Designers have explored the automotive environment using a variety of design research tools. Designers often interview people using speculative participatory design methods (Pettersson and Karlsson, 2015) or have them fill out diaries after a drive to understand the in-car experience, for example, with music and sound in the car (Bull, 2004). These *post facto* methods can provide rich depictions of user experience, but limit designers to moments that the users noticed and remembered. Incorporating in-vehicle video recording of naturalistic driving allows for post-drive video analysis, but researchers often lose the ability to inquire about what is being thought about or observed in-the-moment. For this reason, Perterer et al. (2013) argue that truly understanding drivers and passengers can only be done by being in the car.

In situ methods such as contextual inquiry, ethnographic study, and cultural probes can be adapted to the constraints of the car; for instance, in-car observation and contextual inquiry have been used to understand the use of GPS navigation systems (Leshed et al., 2008) and to test the design of new collaborative navigation applications (Forlizzi et al., 2010; Gridling et al., 2012). During these studies, the experimenter can observe the real-time experience of drivers and interact by asking questions. However, Meschtscherjakov et al. (2011) note that the presence of the researcher in the car presents its own problems. For instance, coordinating a researcher to ride-along can be difficult and can limit drives to planned routes, and cut down on the spontaneity of naturalistic driving. The presence of the researcher can also bias the behavior of participants and potentially influence a driver's actions or make a driver feel uncomfortable if they do not know the researcher. It can also be challenging for the researcher to observe the situation at a high-level, take detailed notes about specific aspects of the drive, and control the behavior of interactive prototypes. Researchers can also become motion sick while attempting to write notes or control screen interfaces in the car. Finally, any interaction within the car can pose a safety risk by distracting the driver. I have developed WoZ Way to address the associated challenges with in-vehicle observation and interaction prototyping while still providing designers a rich view of the on-road experience. The system employs remote communication technology with a Wizard of Oz methodology to allows a researcher to "be there" without actually needing to be there.

4.2.2 Wizard of Oz Prototyping in Driving Studies

Originally developed by J.F. Kelley to prototype an interactive, speech-based computer calendar service in 1983 (Kelley, 1983), the Wizard of Oz (WoZ) methodology has long been used to stand in for computational recognition, planning, and interaction systems that do not yet exist. From a design perspective, WoZ enables a fluid and improvisational means to test new design ideas and to understand people using the technology in specific contexts (Dow et al., 2005b). With well-designed WoZ systems, non-technical designers can engage in rapid interactive prototyping of the behavior of new interfaces (Klemmer et al., 2000) and explore new interaction possibilities between people and technology (Molin, 2004).

WoZ methods have long been used in automotive interaction design to simulate and explore systems requiring considerable hardware for sensing or processing. Often, these experiments involve driver interfaces in on-the-road test vehicles which were surreptitiously controlled by a back seat experimenter with a numeric keypad (Green et al., 1993a,b; Green, 1995) or tablet PC (Hogema et al., 2009). WoZ techniques are also frequently employed to develop speech interfaces for cars (Geutner et al., 2002; Lathrop et al., 2005) and collect data for bootstrapping natural language algorithms (Dahlbäck et al., 1993; Weng et al., 2006). WoZ speech systems have also been used to explore user experience. Sirkin et al. (2016) used WoZ speech in a driving simulator to promote ad hoc conversation with drivers. This setup enabled researchers to inquire what people were thinking and feeling while minimizing the distorting effects that can occur when people are being interviewed in person, a common issue for usability researchers (Weilenmann, 2001).

Recently, WoZ has been used to simulate wholly autonomous vehicles. Schmidt et al. (2008) modified a vehicle to have a hidden compartment with driving controls for the Wizard. This allowed the Wizard to fully simulate and prototype the behavior of advanced driving features such as automatic lane keeping and infotainment interfaces. Furthering this work, the Real-Road Autonomous Driving Simulator (RRADS) system utilized both a hidden Driving Wizard to simulate autonomous driving and a known Interaction Wizard in the back seat to control various car interfaces (Baltodano et al., 2015). While these studies have shown the benefits of the WoZ methodology for on-road interaction studies, they can require significant modifications to keep the Wizard hidden in the car and to preserve ecological validity. To avoid these challenges, we have taken inspiration from the ubiquitous computing community to create a system that allows remote WoZ interaction and observation. In doing so, we allow designers to experiment remotely with driving experiences in an ecologically valid manner, an important aspect of Ubicomp studies (Carter et al., 2008).

4.2.3 Remote Observation and Interaction Prototyping

Much of the prior work in remote observation techniques for designers to understand users comes from the mobile phone space. Early systems focused on extending the experience sampling method (Csikszentmihályi et al., 1977) as a technique to evaluate and improve Ubicomp applications (Consolvo and Walker, 2003). Froehlich's MyExperience (2007), for example, enables designers to survey users after they completed specific interactions with their device. One example had users rate the call quality after a mobile phone call on the phone itself. Carter et al.'s Momento (2007) was developed to help designers better understand the in-world context in which people were using mobile computing applications. Designers receive notifications around specific trigger events notifying of a participant's mobile application use, in real-time. They can subsequently interact with the user over multimedia messaging by sending questions and requesting photos or videos of the user's environment.

While the discrete messaging model of Momento worked well for the incidental use patterns typical of mobile applications, higher bandwidth observation and interaction are required to enable continuous and sustained experiences. McIntyre et al.'s DART (2004) allows designers to prototype augmented reality experiences in real-time, responding to interaction data transmitted over a network. In one implementation, the designers prototyped a "voices of the dead" storytelling experience of the Oakland Cemetery by following participants from a short distance, logging their GPS coordinates, and triggering audio on their AR interface (Dow et al., 2005a).

In discussing the future of remote interaction design tools, Crabtree et al. (2006) point out that many aspects of interaction in ubiquitous environments are invisible and fragmented, and that "there is a strong need to enhance observation in these environments, making the invisible visible and reconciling the fragments to permit coherent description." They champion the combined use of video and system log data in remote ethnography to enable sensemaking of increasingly computational user experiences. Most recently, Chen and Zhang created a system using Google Glass and video conferencing to remotely paper prototype mobile applications (Chen and Zhang, 2015). This system allowed a designer to quickly test low-fidelity prototypes in the wild, helping them discover issues they could have never found in the lab. The WoZ Way system blends the video-based WoZ capabilities of this remote paper prototyping with the data and interaction features of Momento and DART for use in the car.

The automotive setting is a strong example of an environment where the challenges addressed by this previous work on *in situ* interaction prototyping are present. Our system goes "beyond being there" (Hollan and Stornetta, 1992) to provide the designer with a holistic view of the visible and invisible aspects of the driving experience.

Designers of in-car experiences need to respond in real-time to events that are occurring in the vehicle, around the vehicle, and throughout the drive. An interaction prototyping system that addresses this context needs to integrate information about the driver, from the car, and from the road in order to be complete. The system needs to be networked to span the distance covered by a drive. We extend previous systems, such as Carter et al.'s Momento (2007), by providing networked, high-quality, real-time video and automotive data with the ability for designers to directly interact with a driver through speech and in-car interfaces. The architecture of our system allows designers to observe and interact with drivers in their remote contexts, providing a platform for understanding and creating real-world driving experiences. WoZ Way is both a design research and a design prototyping tool.

4.3 System Description

Using WoZ Way, researchers can observe and interact with drivers during their everyday commutes, reducing the logistical struggles of route planning, enabling on-road observation, and allowing real-time prototyping of in-car interface behavior.

4.3.1 Function

WoZ Way allows designers to watch the real-time driving experience via high fidelity video and audio, and also simultaneously receive meta-data about the drive, such as the vehicle telematics data and real-time map or traffic information. The designer can also interact or intervene, asking questions by using a text-to-speech messaging system, or remotely triggering custom in-car screen and electromechanical interfaces. Together, this observation and interaction allow the designer to explore and experiment with the driver's experience.

4.3.2 Key Features

The key features of the WoZ Way system include:

- high bandwidth, low latency, self-resurrecting remote connection capabilities
- real-time synchronization of multiple data streams from the car to the designer
- "interaction through a machine" capabilities using multi-language text-to-speech and remote control of electromechanical interfaces to prototype their in-car behavior
- multi-channel data capture to enable *post facto* analysis of interactions

Observation

By having a remote connection to the car, a designer can understand experience without physically riding along. This allows for exploration in real-world contexts and shows the designer a richer picture of someone's experience in the car. Live video and car data streams provide the remote designer more information than what is available to people sitting in the car. With these data streams, designers can take a holistic view of the visible and invisible actions in the car. They can observe facial expressions, system usage, and on-road events within a single interface. This helps to bring the data-driven methodologies used for understanding people on the web into the physical world.

WoZ Way also allows researchers to be a part of longer one-way drives that would otherwise be inconvenient for the researcher to join, such as a long commute. The mobile nature of the system also allows observation during multiple drives over many days, allowing more longitudinal studies of drivers. Finally, it increases the ability to observe and design services that address passengers, such as children being dropped off or picked up from school or group carpooling.

Data Streaming and Capture

WoZ Way has also been designed to provide the designer with more contextual information than they could potentially see if they were merely riding along in the car. Live streams of car data and multiple camera views give multiple perspectives on the same scenario. This not only gives the designer a richer picture of the driving experience but also helps alleviate the workload to record and respond to low-level details and overall experience. Automatic collection of video, audio, and automotive data also allows designers more freedom to interact with a driver and can be used for more detailed post-drive analysis. Researchers will also avoid motion sickness as they are not writing notes or controlling a computer interface inside of the car. This can allow for safer testing as the designer can focus on their interaction with the driver rather than on the mechanics of the study.

Interaction Prototyping

Interaction between the designer and participant is enabled through in-car speech, screen, and electromechanical interfaces. Designers can prototype the interactive behaviors of in-car interfaces by interacting through a machine. This allows designers to use connected technology as a design tool to understand people. Additionally, by being the voice and behavior of the machine, the designer simulates the mind of the machine, helping to draw out a sense of the appropriate "goals, plans, expectations, and desires" for future technologies (Norman, 1993). In some ways, WoZ Way allows the designer to hide, even when interacting with people in the car, by allowing him or her to act as the car. This can help to avoid the unintended biasing pointed out by Meschtscherjakov et al. (2011).

Safety

Safety is a key system design consideration for WoZ Way, as it is meant to be used on-the-road with actual drivers. One advantage of the WoZ Way system is that the Wizard can observe both the driver and driving context in real-time. Having the remote observers see what the driver sees on the road helps to mitigate distraction (Gaspar et al., 2014). This allows the Wizard to be judicious in choosing when to interact with the driver, or when to control in-car interfaces responding to the driver or environment. It is important to note that having a view of the road is critical. A video chat of the driver or voice only is can lead to similar problems as cell phone distraction (Charlton, 2009). WoZ Way also allows the Wizard or a supervising researcher to turn off any interactive prototype during an on-road test easily. This ability for the Wizard to fully control or disable interactions with the driver is an important capability that is not present in current-day in-vehicle interfaces. In fact, the information about when the Wizard chooses to interact and when the Wizard chooses to wait could be an important input into how future in-vehicle systems interact with drivers.

While it is possible to prototype the behavior of many systems in the car, special care must be given to what interactions will be explored. For example, it is relatively easy to connect to and control vehicle subsystems such as information displays, steering, braking, and acceleration via the CAN interface (Miller and Valasek, 2013). In some cases, these may present interesting opportunities for prototyping the car's behavior and may be warranted for some studies. For example, Jung et al. installed and tested custom battery indicators to explore their impact on driver range anxiety in electric vehicles (Jung et al., 2015). Researchers have also designed hidden driver compartments allowing an experimenter to prototype features such as lane keeping (Schmidt et al., 2008). While WoZ Way could allow for the remote control of these systems, for safety reasons, we have focused our interaction prototyping on speech interfaces and brought-in electromechanical interfaces that do not interfere with the driver's ability to control the car.

4.3.3 Architecture

The system architecture for WoZ Way has three major components as seen from left to right in Figure 4.1:

1. A Wizard interface with live video, audio, and data displays from the remote vehicle and controls to send text-to-speech messages to the driver or control the behavior of in-car prototypes

2. A mediating data server to manage communication between the remote vehicle and the Wizard interface; the data server also collects time-stamped data logs
3. A computer in the car to collect and share video, audio, and automotive data over the internet with the Wizard interface and to control spoken text-to-speech messages, screen interfaces, and electromechanical components
4. Auxiliary interfaces including screens, sensors, and actuators used in each specific study

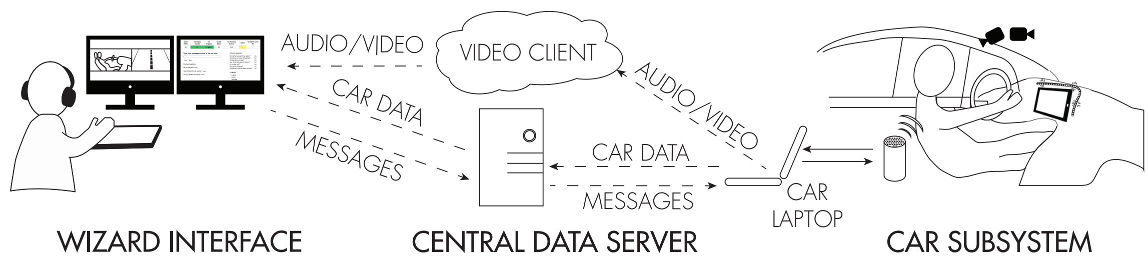


Figure 4.1: WoZ Way System architecture for on the road, real-time, remote observation and interaction prototyping.

The components of this architecture allow for data to flow between the remote car and Wizard. Video and audio are streamed from the car using a video chat client. Car data and interface control messages are streamed through a separate, centralized data server. The Wizard interface shows the data streams from these two sources and allows for creating custom device controllers.

4.4 Implementation

To encourage adoption of WoZ Way as a research platform for in-vehicle experiences, we implemented the majority of the WoZ Way architecture using off-the-shelf hardware and software. Custom software for the in-car data and interaction systems and the Wizard control interface is written using widely available open-source tools and is made available on GitHub for others to modify and reuse.[2]

[2]https://nikmart.github.io/WoZ-Way/

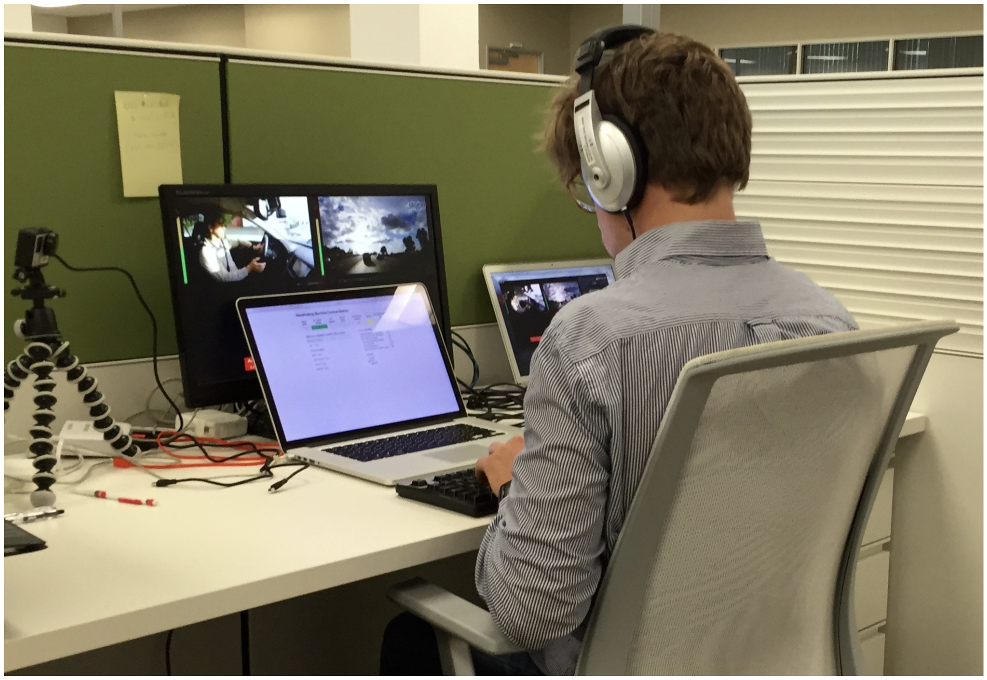


Figure 4.2: WoZ Way control station with external HD monitor for video and digital control interface and data viewer.

4.4.1 Wizard Interface

The Wizard interface, which is based on HTML and JavaScript, can be adapted for different applications and deployments. In general, it useful to have the Wizard interface divided into three main regions: The *display area*, which features the live video, audio, and car data feeds; the *Wizard input area*, which allows the Wizard to input text and track the history of queries; and the *control area*, which features buttons for settings, common text-to-speech messages, and custom controls for auxiliary interfaces.

When possible, it is best to dedicate a separate high-definition monitor for the *display area*, giving the Wizard a large view into and around the car, and leaving more room for the interaction controls, shown in Figure 4.2. Figure 4.3 shows a detailed view of the interface that was used in one of our test deployments focused on understanding the driver experience of automatic cruise control systems. Live data from on this interface includes:

- **Vehicle speed [mph]**

- **ICC Display Engaged** – shows if the ICC system is ready and the display on the console is visible to the driver
- **ICC Engaged** – shows if the ICC has control of the vehicle speed, braking, and lane keeping
- **ICC Set Speed** – speed setting for cruise control in [mph]
- **ICC Following Distance** – setting for how much space to leave ahead of car [`long, medium, short`]
- **Brakes** – indicates if brakes are engaged [`1`] or off [`0`]. If brakes engage and **ICC Engaged** stays `ON`, the car is automatically braking. If brakes engage and **ICC Engaged** turns `OFF`, driver is braking manually.
- **ICC Speed Above Set** – indicates if car is traveling at or below set speed [`false`] or faster than set speed [`true`]. If [`true`], driver may have accelerated manually.

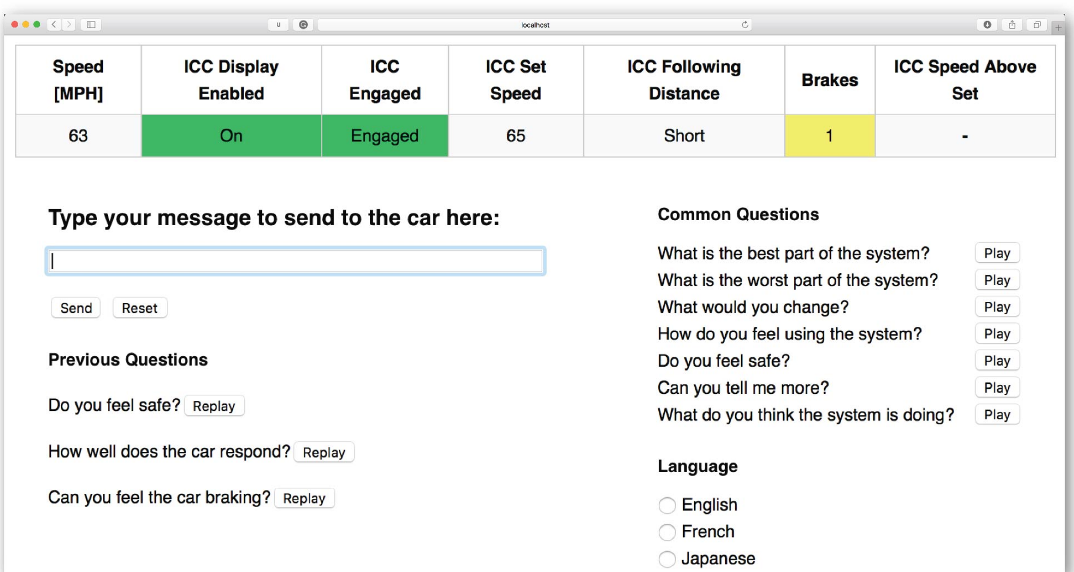


Figure 4.3: Wizard control interface and data viewer showing live automotive data and input for speech interaction.

Vehicle speed is updated at a rate of 1 Hz. To minimize network bandwidth, transmission of all other data measures are triggered by change events. Color and text are used to encode data values and allowed the Wizard to quickly see changes in the car's state. It should be noted that this information is not available on all cars and may not be required for all studies. Thus, each interface should be designed with the study goals in mind.

In the input region of the interface, we provide a text input field where the Wizard can send messages to be spoken or displayed in the car. Messages are sent on an `enter` key press or using the `send` button. Sent questions are logged sequentially below the text input field. Each sent message has a `replay` button to quickly repeat the message.

The control region of the interface features a set of common questions or interface controls. A set of programmed messages enables Wizards to respond quickly to common and anticipated events. We also use this region to enable settings, such as the language used by the text-to-speech system, as well as controls for in-vehicle interface prototypes. For example, in a prototype exploring ambient car lighting, we included an RGB color picker that allowed the wizard to try how new colors were received.

Using the Wizard interface, designers can observe the data stream and live video from the car. While observing, they can actively query the driver using the text-to-speech messaging system or interface messaging system. With added controls, the Wizard can prototype screen or electromechanical interface interactions.

4.4.2 Mediating Data Server

A centralized data server is used to manage communication between the Wizard and car. All car data and input messages are routed through this server so that the Wizard interface and car only need to communicate with the data server. This allows the Wizard station to be used in different network locations without reconfiguration. We use MQTT [3] is used for lightweight machine-to-machine communication. The central data server (Lenovo ThinkServer, Ubuntu 14.04 LTS) runs a Mosquitto MQTT broker.[4] The server is located on a university campus and networked with a 1 Gbps Ethernet connection. In addition to managing the communication, all messages to the server are time-stamped and logged for

[3] www.mqtt.org
[4] http://mosquitto.org

later review.

4.4.3 Car Subsystem

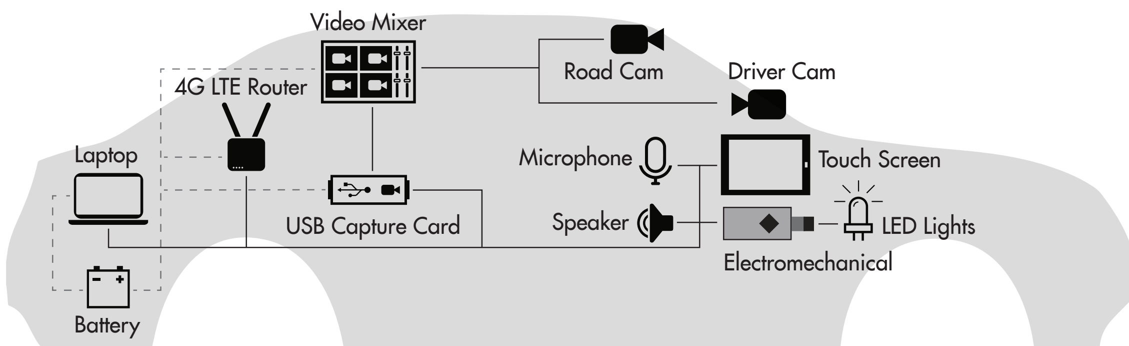


Figure 4.4: WoZ Way in-car subsystem which allows for remote control of speech and electromechanical interfaces along with live video and audio.

A laptop computer is used inside the car to manage video, audio, and car data capture as well as to generate the text-to-speech messages received from the Wizard station. This computer also mediates communication to auxiliary interfaces. High speed (5 - 10 Mbps) internet is provided by a 4G LTE Wireless Router (Cradlepoint COR IBR600).

Video and Audio Streams

Our system implementation supports up to four video stream inputs, including video capture and screen capture. For most projects, we have found it necessary to have one view that includes the driver's hands, facial expressions, and body posture, and another that includes driving context through a road view. We usually have one or two further feeds dedicated to specific information such as the instrument cluster, a facial close-up, or an interface screen capture.

The video feeds are connected to a video multi-viewer (Gra-Vue MIO MVS-4HDMI), which synchronizes and stitches the video into a single 2x2 view. This stitched view is then connected to the computer via a video capture card (Inogeni 4K2USB3). An example of the video feed is shown in Figure 4.5. Audio is captured using a high-quality microphone (CAD

Figure 4.5: An example of a multichannel video stream from in the car during a WoZ Way session.

Audio U9 USB Condenser, Omni) mounted to the rear view mirror and directed toward drivers, to capture his or her speech and minimize road noise inside the car. The video and audio are streamed using a video chat client. We specifically chose Skype, as it can stream high quality (720p) video and audio, and dynamically prioritizes audio quality over video quality. Additionally, Skype is configured to automatically connect calls and start streaming video and audio without ringing, which allows automatic connection resurrection following inevitable on-road network disconnect events.

Car Data Stream

Live car data such as speed, braking, and system status is captured from the car's built-in On-Board Diagnostic (OBD) port. We capture data using a Bluetooth OBD logger (OpenXC compliant CrossChasm C5), which gives access to low-level, manufacturer specific data streams often not available in other OBD loggers. This device also allows for custom data filters and rate settings, allowing us to limit the amount of data captured to what we want to provide to the Wizard. The data is streamed over Bluetooth to a custom program running on the car laptop.

Input messages

Text-based messages are sent from the Wizard station to the car and are rendered to audible speech or on-screen messages by an in-car laptop. For the text-to-speech system, we leverage the built-in speech capabilities of MacOS. Sounds and spoken words are communicated through a wired, portable speaker (JBL Charge 2) placed in the car's cup holder. Using a separate speaker allows drivers to listen to music or use the in-car navigation as they normally would.

Auxiliary Interfaces

The Wizard can control custom interface prototypes using control messages sent from the Wizard interface via the laptop over WiFi. Screen-based interfaces can be prototyped using a tablet computer attached to the center stack of the car. The tablet also provides a built-in set of sensors such as GPS, acceleration, and orientation that can be streamed back to the Wizard interface, if desired. For custom electromechanical interfaces, such as ambient light controllers, we use microcontrollers with a USB serial interface to communicate data back and forth between the interface and Wizard control station.

4.5 Test Deployments

In order to explore the use of WoZ Way in real design contexts, we conducted three test deployments with designers and on-road drivers. In each of the studies, the designers acted as Wizards to interact with the drivers. Our tests include two proof-of-concept deployments, where we piloted novel in-car prototypes: 1) an interactive chatbot that converses with drivers during commutes, and 2) a touchscreen-based application of the car's center console. In our third deployment, we collaborated with an automotive industry research lab on an *in situ* field study exploring the driving assistance of advanced Automatic Cruise Control (ACC) and lane-keeping systems. In the next sections, we describe how designers and researchers employed WoZ Way for various purposes across these test deployments. These serve as examples of how the WoZ Way system can be used as a tool for observation and interaction prototyping.

4.5.1 Protocol

During all driving tests, the Wizards were free to interact with the drivers as they deemed appropriate for the specific test deployment. Drivers were made aware that a person was controlling the interfaces. After every drive, we interviewed the Wizard for his or her experience using the system. We also interviewed the drivers, when available, about their experiences interacting with the speech system. We recorded video of the drivers, Wizards interacting at their stations, and screen captures of the Wizard interface. We also recorded audio of post-session interviews. These deployments were conducted under a research protocol (IRB-36970 Understanding on-road driving assistance) approved by our institutional review board.

4.5.2 Interactive Chatbots

For one test deployment, I collaborated with Pablo Paredes who is exploring the commute as a space for mental wellness interventions. For one prototype interface, we developed a speech-based chatbot for conducting therapeutic interviews with drivers. Drivers interact with the chatbot via speech while driving on their daily commute. The chatbot is controlled by a Wizard with experience conducting wellness and therapy interviews. WoZ Way is set up as the prototyping platform for testing interactions between the chatbot and driver, as well as a tool for collecting data to train natural language models for car-based wellness conversations. We modified the audio setup such that the speech comes from the car's speakers and appears as if you were chatting with the car itself. We deployed the system to test its appropriateness for speech-based interaction prototyping during a short 10-minute test drive. The interface for this setup is shown in Figure 4.6.

The Wizard and driver framed their discussion about food choices. During the the drive, the Wizard responded to the driver's answers and asked different types of questions to delve deeper into topics such as the driver's reasons for skipping meals, motivation to work hard, and goals in life. Our goal during the session was to explore how well the system would allow a chatbot designer to interact with the driver.

4.5.3 Context Enabled Touchscreen Interfaces

In another research project, I collaborated with David Sirkin who was developing a touchscreen interface using a tablet computer to assess the Situation Awareness (Endsley, 1995) of drivers on the road (Sirkin et al., 2017). During the drive, the Wizard controls pop-up

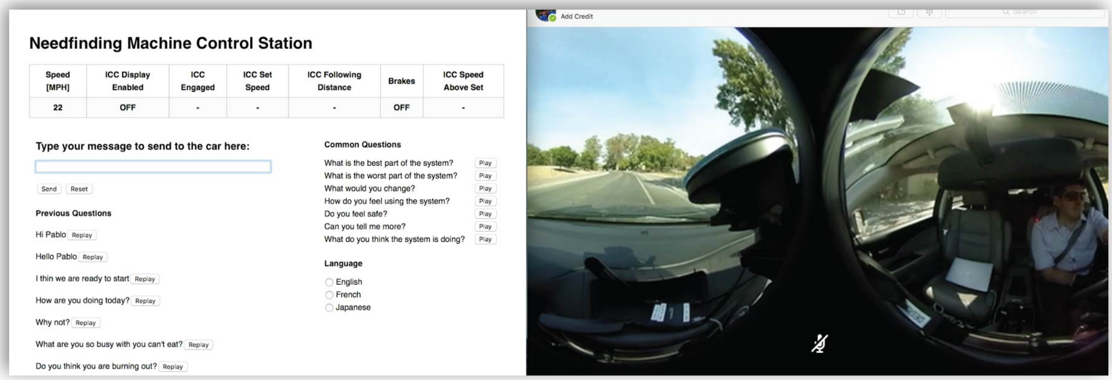


Figure 4.6: The chatbot control interface used for testing in-car speech agents for mental wellness.

messages that ask drivers questions about events that occur on the road, to see how well they notice their surroundings. For example, the Wizard might ask drivers if they passed a bicyclist or construction zone. For this project, we also developed a simple set of ambient lights around the dashboard, to alert the driver that a question was waiting on the center console interface. These lights turn on when a new screen alert pops up and can glow specific colors determined by the Wizard, shown in Figure 4.7.

Our goal in this deployment was to test the ability of a remote Wizard to control interactive screen and electromechanical prototypes while also conducting a user test of a new application. We used the WoZ Way system to allow designers to observe the drive and to act as system Wizards to dynamically respond to on-road events without interfering with drivers. In a previous prototype of the situation awareness application, a designer sat in the back seat of the car to observe. While this setup works, the presence of the designer may heighten the driver's awareness. Additionally, some of our researchers are prone to motion sickness while working on a laptop interface in a moving car. We modified the Wizard control interface for the current study to allow the Wizard to see the road, the driver's face, the driver's upper body and arms, and a screen capture of the application. We also added common alert messages with associated colors so that the Wizard could quickly ask if the driver saw common events like pedestrians. We used the text input area to allow the Wizard to create custom messages for unique events on the road. Finally, we added an RGB color

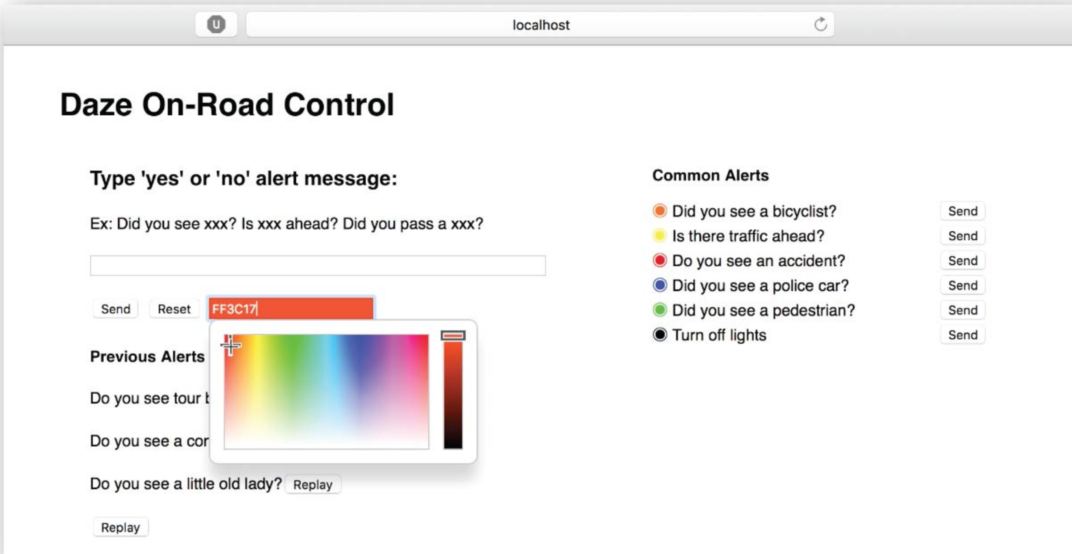


Figure 4.7: The touchscreen application message and light remote control interface used for a situation awareness study.

picker to allow the Wizard to change the ambient light colors with relative ease.

During a 30-minute pilot, a Wizard spoke to a driver through the video chat interface to provide directions and to inform the driver about the situation awareness application features. The Wizard opted to speak directly to the driver, as it was easier to direct the driver along an ad hoc route. The Wizard followed the driver's location on the map and observed the surrounding area to choose locations for the driver to head based on where he thought interesting on-road events would occur.

4.6 Understanding Currently Available ADAS with Renault

While the previous two test deployments were developed with design researchers working with our lab group, we also wanted to see how industry automotive designers and researchers could employ WoZ Way in the context of a real design challenge at an automotive research facility. Our partners are currently studying drivers' use of advanced cruise control (ACC) and lane-keeping systems. The designers were collecting real-world usage logs, but they were interested in understanding drivers' in-the-moment experience. In addition, the group had an interest in designing speech-based interaction systems for use in the car. We co-developed a driving assistance case study to allow these researchers to observe and interact with drivers remotely. As the designers were primarily focused on understanding the current driving experience, this case study focused primarily on speech-based contextual inquiry. It serves as an example of the rich, ethnographic style of research that can be conducted using the WoZ Way system.

Vehicles Used

The study used four vehicles, across 12 different interactions. We conducted 10 sessions in either a 2016 Infiniti Q50 or Q50 Hybrid with Intelligent Cruise Control (ICC). Two pilot sessions were also conducted, one in a 2013 Tesla P85D with Autopilot and one in a 2016 Nissan Murano with ICC. The ACC+lane-keeping capabilities and the control interface of the hybrid and non-hybrid Infiniti Q50s were identical.

Wizards and Drivers

We recruited nine Wizards from the research facility staff. During the two pilot sessions, the first author acted as the Wizard. For the 10 sessions in the Infiniti Q50s, the Wizards were either designers, user researchers, or engineers currently working on aspects of new driving assistance features and in-car interfaces. All Wizards were interested in understanding the driver's experience using the ICC systems. Five Wizards had formal training with user research methods and have conducted formal user studies in that past. One Wizard participated in two sessions.

We also recruited 11 drivers from the automotive research facility to take test drives in the vehicle. These drivers were engineers, designers, and researchers working on new automotive interface and driving assistance systems. One driver participated in two sessions: one in English and one in French.

Driving Sessions

Drivers were invited to borrow one of the vehicles to either take home for the evening on their regular commute or for a 30–90-minute test drive. Drives ranged from 10–60 miles. Before their drive, drivers were told that the purpose of the session was to understand the driving assistance using the ACC+lane-keeping systems available in the car. They were told to drive safely, and as they usually would, but also to try the ACC+lane-keeping system when they felt it was safe to do so. Drivers were allowed to listen to music, use their phone, or have a passenger ride along. A passenger accompanied two of the drivers.

Wizards interacted with the drivers through the text-to-speech system, asking questions about the drivers' experiences. The Wizard interface was modified for this study to include a live data stream including ACC+lane-keeping system state. We worked with our industry partners to develop a set of standard design research questions to such as "*Do you feel safe?*" or "*Can you tell me more?*" We also modified the speech system to allow French and Japanese researchers and designers to use the system.

4.7 Findings

In this section, we discuss how the designers and researchers across all of our test deployments used the remote observation and interaction system. After each driving interaction, we conducted semi-structured interviews with the Wizards, and when possible, the drivers, about their experience using the system. We reviewed the interviews and have grouped our findings into four categories: system benefits, challenges, methodological discoveries, and implications for ethnography and design.

4.7.1 System Benefits

Reducing Interference with the Driver

We found that Wizards could react to on-road events that they observed without interfering with drivers. For example, during the touchscreen interface deployment, Wizards were often nervous about drivers' behaviors, much like a backseat driver might be. Since the

Wizards were not physically present in the car, however, these actions did not influence or bias drivers' actions.

We have a similar anecdote from the driver's perspective during the chatbot deployment. The driver noted that talking to a machine voice allowed him to focus more on the road rather than needing to turn and acknowledge the person sitting next to him. This allowed him to speak more freely without his need for visual, social feedback from another person.

System Supports Flexible Use

Wizards were able to adapt WoZ Way to their own research goals. During the chatbot test, the researcher felt that WoZ Way allowed him to quickly explore the interaction possibilities of a chatbot in the car. He found the system useful for generating hypotheses for future studies and for generating ideas for autonomous chat algorithms.

The researcher in the touchscreen interface deployment found the system to be useful both for exploring how the application could interact with drivers and for running semi-controlled field studies.

In our experiences with industry designers, we found that Wizards with different backgrounds within the company used the system to help them answer specific questions that related to their work. The system also allowed designers without formal user experience training to engage in exploring the driving assistance features. Designers with backgrounds developing control algorithms for advanced driving features noted how watching the interaction made them interested in making the automatic driving experience more human. These system designers often asked questions regarding the feel of the automatic speed adjustments during events such as lane changes or low-speed driving. Other designers working on in-car displays asked questions about the instrumentation in the car, what information the driver understood, or how the driver interacted with components such as the entertainment system. For example, when one driver turned on the radio, the Wizard asked, "*What do you listen to: radio, podcasts?*" Other researchers who were exploring interactions outside of the car even asked questions about how the driver interacted with bicyclists and pedestrians. Overall, the system allowed designers and researchers with varying backgrounds to investigate driving assistance and test new interaction prototypes.

Designers Improvise New Services On the Spot

We found that designers and researchers in the driving assistance deployment responded to drivers' experiences by prototyping new interactions as they came up during the drive. Some Wizards experimented with using WoZ Way to prototype an interactive user support system. For example, researchers who worked on advanced driving systems helped the drivers understand how the automated system worked when the drivers were confused. In other instances, drivers would request information from the Wizard about things relating to the drive itself. In one vivid example, a driver using GPS to get to a university campus was unable to find parking. The Wizard was familiar with the campus, and guided the driver to one of the visitor parking lots with turn-by-turn directions, using the live video feed to determine the car's location. The Wizard was even able to pay the parking fee from a mobile application on their own phone. From this emergent experience, the designer was able to prototype the interaction for a parking spot assistance system, a feature currently absent from most navigation applications.

Synchronized Data Streams Reduce Designer Workload

When asked how they used the Wizard interface, many Wizards found the video and data display to be useful for understanding the car's state. They remarked that it was nice to get a sense of what drivers might be able to see on the road and their dashboard. By having all the video, data, and controls in one interface, the Wizards felt they could control the inside of the car while experiencing the drive from the driver's point of view. Having data automatically synchronized and logged freed the Wizards from spending time to establish a ground-level understanding of what was happening, like "*Is the ICC on?*" Wizards focused on asking higher level questions, such as "*Do you feel the ICC is more likely to fail with heavy traffic?*" or "*Do you feel the car prevented you from driving how you normally would?*"

Interaction Alleviates Concerns Over Surveillance

During our evaluation, the drives were explicitly framed as user experience studies. All the drivers knew that there was a person on the other end whose goal was to test new automotive user interfaces or to understand the driving experience. Even after understanding the data recording systems used in the car, most drivers did not voice concerns about privacy or being surveilled. If drivers did not want to answer a question, they seemed comfortable to stay silent. However, interactions were generally viewed positively, and drivers answered questions openly. For example, during the chatbot deployment, the driver stated that speaking with the machine voice made him forget the person on the other end, and caused him to open up about his answers to the interview questions. Another driver in the driving assistance deployment said that the interaction he had, where the machine would respond to events on the road and inquire more about the driver's opinions, made him feel that his feedback might be heard by someone who may actually be able to change the experience for the better.

In another session, the driver initially showed hesitancy in being recorded, asking "*Are you guys watching me?*" The Wizard replied, "*The cameras enable my contextual reasoning.*" This ended up being a very chatty session. After becoming more comfortable with the interaction during the drive, the driver needed help parking, and the Wizard assisted. The driver even began to thank the system for helping her navigate to a location she was unaware of. This provides an example of how interacting with drivers, rather than just monitoring them, can go beyond disclosure to help ease privacy concerns; the interaction helps to frame the purpose of the data system as a collaborative design tool rather than as spyware.

Remote Interaction Reduces Logistical Headaches

During our drives, we avoided the logistical challenges of driving along with people on their daily commutes. Although setting up the system can take some time, it gives Wizards more freedom to be a part of the drive from their own location. Many of our drives began or ended at drivers' homes, sometimes up to 60 miles from designers' offices. Using WoZ Way allowed designers to observe and interact in these everyday drives without requiring the complicated planning associated with getting researchers to meet up at someone's home, or arranging chase vehicles to take researchers back home after the fact. This reduced logistical burden made it possible to do more runs and observe more people.

4.7.2 System Challenges

Dangers of Remote Interaction

Wizards across all the test deployments felt that they needed to alter their behavior based on what drivers were doing on the road, so they would not distract the drivers too much. Specifically, during taxing or dangerous events for drivers, Wizards often held off on planned interactions. They were often challenged to find the appropriate time to ask about the experience. One Wizard noted after a cut-off event that she tried asking a question when the driver was overloaded and needed to wait until the driver had calmed down to ask the question again. Although the driver was able to ignore the question and focus on driving, the Wizard would have liked to understand what moments were, and were not, appropriate to interact with the driver. We also saw this in our interaction prototyping deployments. Wizards would often cringe if they sent something at the wrong moment as they were asking questions with the chatbot or controlling the touchscreen.

Throughout our testing, no prototyping or observation sessions needed to be stopped. While drivers did note that some interactions from the speech system were awkward, these did not inhibit the driver from safely operating the car. Designer's who had an awkward interaction often waited for a better moment and then continued their testing. Still, this suggests further research about how designers should handle these situations.

WoZ Way is Network Dependent

Wizards generally did not notice significant latency from the time they sent a message to hearing it or seeing an interface change in the car (< 1s). However, in two sessions, heavy network loads caused the video and data link to be severely delayed (> 5s) or to completely disconnect for several minutes. This was associated with dense, slow-moving traffic; it is possible that cellular towers are impacted by having too many cars in one location. When the system is fully disconnected, there is nothing the designer can do but wait for a better signal and for the car to reconnect. This causes gaps in real-time understanding of the driver's experience, loss of control of the in-car interface, and can cause a breakdown in the interaction, as the driver is often unaware the system has disconnected. During sessions with poor video quality or a loss of car data, the Wizard often reverted to asking low-level system questions, like what speed the driver was going. During one session with very poor video quality, the Wizard had the driver conduct a speak-aloud protocol to describe what was happening. Although this worked for one session, it speaks to the importance of having high-quality, real-time data streams and for designing robust systems with automatic reconnection.

4.7.3 Methodological Discoveries

Remote Observation is Different from Backseat Observation

Although designers are virtually riding along with drivers, we found a number of differences from having designers in the car with drivers. Designers were better able to understand the single-driver experience. Remote designers also had a better view of the driver than is available from the passenger or rear seat without the awkward overhead of trying to remain invisible while being hosted as a guest of the driver. The researcher in the touchscreen interface deployment stated that he "*felt like a drone pilot or an air traffic controller.*" The ability to see many views and data streams at once was something he could not do before.

Interacting remotely also changed the process used by the designer while observing and interacting with drivers. A design anthropologist in the driving assistance deployment noted how she did not feel the need to make small talk, asking about the driver's day or her plans for the evening. Instead, she either asked focused questions or sat back and watched the drive in silence, observing the driver and the surrounding context without interruption. The Wizard felt that this allowed her to think more about what interactions were happening rather than thinking of what to say next. This allowed her to more readily understand and engage with the driver's on-road experience.

From the drivers' perspectives, their interactions with the Wizards through the machine interface gave them more freedom not to answer questions. For example, during the driving experience studies, some drivers did not answer questions because they were focused on the driving task. Other drivers did not respond to questions they were not comfortable answering. Still, many drivers forgot that the car interfaces were controlled by a person and interacted as if it were a machine. The removed social presence of the designer gave more room for the drivers to act more naturally while driving.

Delays Disrupt Fluency but Increase Disclosure

Although the relay of messages or control commands was near real-time, Wizards were often limited by their typing speed during speech-based interactions. This altered the interaction between the Wizards and drivers. During the chatbot and driving assistance deployments, Wizards discussed how they would begin writing a new question only to have the driver change the topic. This caused the conversation to lose fluency as the Wizard needed time to write a new question. Drivers noted how long pauses broke the flow of conversation. However, this also made them feel more like they were talking to a machine. Often, the long silences led the drivers to feel as if they should expand more on what they were talking about. Although this is partly a challenge of using a typing-based system, we found that the natural elicitation from awkward silences could be led to deeper discussions.

4.8 Discussion

Based on the development of WoZ Way and the evaluation with designers and researchers, we believe our system enables many new opportunities for remote observation and interaction during the development of interactive automotive technologies. During the deployments, these professionals were able to explore a wide variety of situated real-world driving experiences. They then employed a variety of improvisational interaction techniques to understand drivers' experiences better. By blending rich, social interaction and web-like data collection, WoZ Way allows for new forms of computer-supported design work for interactive, physical systems in-the-world. This contingent interaction model can support many aspects of the interaction design process.

4.8.1 Implications for Ethnography

Our test deployments captured all the spontaneity and messiness of real-road drives. They included drivers being cut off, getting stuck in traffic jams, listening to music, interacting with passengers, and getting lost. The richness of these real-world occurrences allowed the designers to see a breadth of driving assistance opportunities that would not have been present in a controlled environment.

The real-time capabilities of the system allowed drivers to respond immediately to quickly changing contexts and inquire about relevant and salient experiences a person might have. During the touchscreen interface deployment, the designer was able to ask about the application usage and get feedback right away, which allowed him to further guide the user test. In another example from the driving assistance deployment, the Tesla P85D driver heard a squeak on the door panel and began prodding the various panels in the car. The remote designer asked, "*How is the build quality?*" to which the driver responded "*Not very good*" and explained how the panels should not make a "*quack*" noise in such a high-end vehicle. This interaction showed how even when the focus of the driving study is to explore the driving assistance capabilities, the Wizard is also able to respond to interesting events in the car. This shows the opportunity for designers to understand the holistic product experience and perform contextual interviews in real-time.

By not being in the car, the designers felt more anonymous and removed, reducing social pressure in the conversation. During the chatbot study, the premise of acting like a chatbot pushed the designer to ask more and more personal questions. We saw a similar pattern in the driving assistance deployment. One Wizard suggested that "*The system could be useful for finding deep needs, or deep desires.*" This Wizard asked questions such as"*What new driving pleasure could we imagine in the future?*" or "*Do you think a machine could be better than man?*"

Another designer found that acting through the machine could allow many designers to interact through the system over time with the same driver. This would allow multiple perspectives while having the speech system appear to be stable across interactions with a driver. The designer also suggested that the system could be used for longer-term studies and that it would reduce the need to plan drive-along style sessions.

Finally, the synchronized data collected from the driving sessions can be used for detailed video interaction analysis. We have created an interactive data analysis interface, shown in Figure 4.8, with the video, car speed, and designer questions. This interface allows designers to review drives by selecting interesting points in the data, automatically jumping to those moments in the video. This allows designers to use the collected data as a means to review the qualitative driver experience.

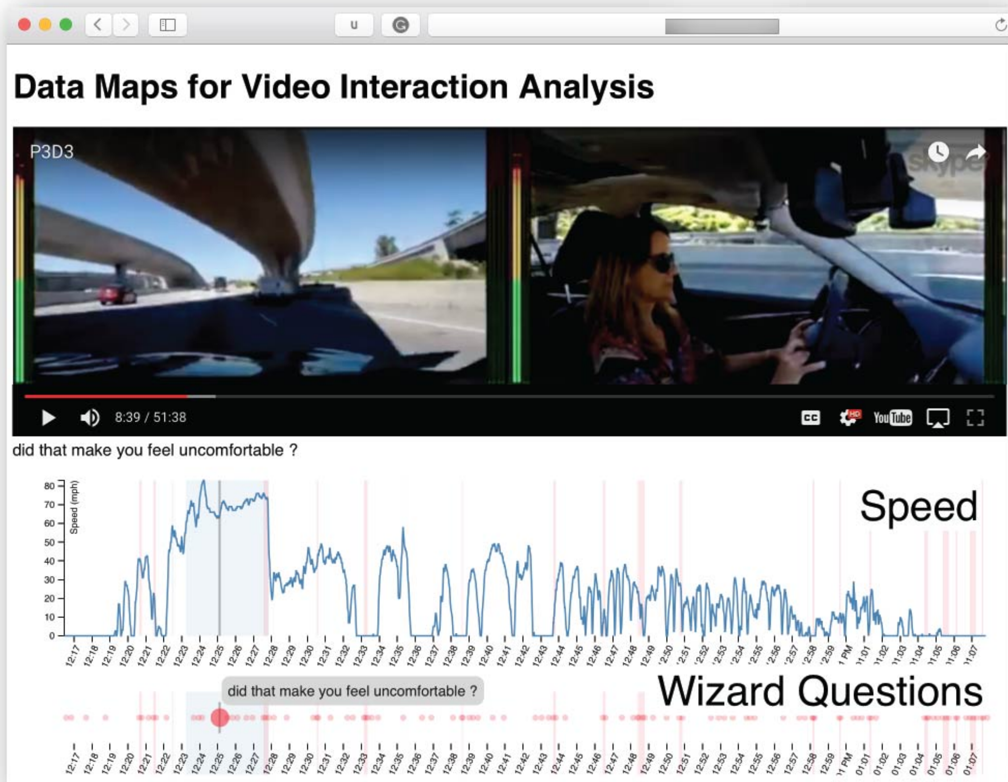


Figure 4.8: A video and data analysis interface created for viewing synchronized WoZ Way data.

4.8.2 Implications for Design

With real-time control of in-car interfaces, designers can prototype new interactions based the driver's behavior and events on the road. The ability to control interfaces remotely allows designers new opportunities for exploring product usage in the real world. The chatbot and touchscreen deployments specifically intended to show the responsive interaction prototyping capabilities of the WoZ Way system. The Wizard in the chatbot deployment found that he was able to actively respond during his interview with the driver. During the touchscreen interface deployment, the Wizard was able to rapidly respond to events on the road and test new interactions with the interface. This is different from testing a programmed application, as it allowed the designer to test new ideas improvisationally. For example, near the end of the drive, the Wizard was interested in how the ambient lights caused the driver to instantly focus his attention toward the touchscreen. He changed the color of the interface without a pop-up message and saw that, indeed, the ambient lights caused a conditioned response for the driver to look at the screen.

One aspect of our system that we would like to address is the difference between interaction and monitoring. As data logging becomes more pervasive in our daily interactions with technology, questions often arise about the ethical use of the collected information. Although our system does allow for remote observation and logging of people's interactions, which can present privacy concerns, we place a focus on facilitating *interactions* between designers and drivers. These interactions reframe the intent of the observation to the design of better product experiences and invite drivers to be active, collaborative participants in the design process. Brereton et al. argue that engagement and reciprocity are critical to conducting ethnographically driven design work, especially with remote populations (Brereton et al., 2014). While their work explores engagements with aboriginal populations, it suggests how we can use interaction to positively frame remote design observation and prototyping.

4.8.3 Opportunities for use in practice

The unique, enabling features of the WoZ Way system can influence the overall design process of interactive automotive systems in many ways.

Evaluating Prototypes

Usability researchers can evaluate prototypes in real-world contexts and gather both qualitative and quantitative data about a person's experience. Researchers can better understand causality and rationale for various behaviors by observing usability data and contextual inquiry.

Designing Interactions

Designers can quickly explore the interaction design space of new technologies without investing significant time toward algorithm development. Interactions can move from improvised to more supervised over time. The underlying computing architecture of the system can easily extend to broader graphical or physical interfaces.

Group Studies

By interacting remotely and not occupying a seat, researchers and designers can better explore how groups interact within the car. This can be useful for studies of families on road trips, coworkers carpooling, and passengers using ride-sharing services.

Mixed Methods Research

The use of WoZ Way enables designers and researchers to employ many types of research methods for their work. The *in situ* observation and interaction provide the designer with rich moment-to-moment understanding. Data logs and video recording from the session then allow for further data and video interaction analysis.

Studying Demographics/Individual Differences

With a remote connection, it is possible to work across demographics and in different locations. With fast internet, the system architecture can be used for conducting studies between different states or countries. For example, we have been able to connect designers in Europe to drivers in California. It is also possible to focus on individual differences between people. This can provide a rich and nuanced understanding of users that can lead to the basis for adaptable systems.

Longitudinal Studies Over Time

It is possible to embed the features for remote observation and interaction into the car itself. More generally, the architecture of WoZ Way is similar to that of connected devices, which is an indicator that the observations and interventions enabled by WoZ Way can be inherent features of connected devices. This can allow exploration of people's experiences over more extended time periods. Since designers simply need to "log in" remotely, rather than be present to interact with someone, they can more readily interact over weeks or months. This can help designers to develop adaptive systems and to see how experiences change over time.

4.9 Conclusion

WoZ Way shows how we can implement and use a needfinding machine in the wild for understanding how drivers interact with systems on the road. The designers in these deployments were able to conduct both contextual inquiry and interaction prototyping in real-time. The designers could see more than they could have if they were riding along in the car. More importantly, the designers could inquire about the driver's experience at the moment that their advanced driving systems were engaged. This allowed them to get the most salient information from the drivers rather than rely on having drivers remember what an experience was like after a drive. By allowing for in-the-moment questioning, WoZ Way allows the designers to begin associating their telemetry data with human experience data. While the tests with WoZ Way show one domain for using a needfinding machine, this model of remote observation and interaction prototyping can be useful in other context-sensitive design spaces.

Chapter 5

Conversations around music

To illustrate another use-case example of a needfinding machine within a specific context, I present the design and test deployment of *DJ Bot*, a remote-controlled speech agent that talks with people to figure out what music to play. *DJ Bot* is a functional system prototype that allows designers acting as remote wizards to play songs and to converse with people about their musical whims and preferences as people listen to music while situated in different environments such as the car or home. I piloted the system and ran a field deployment with interaction researchers at Spotify, a commercial music streaming company. These practitioners were interested in exploring future interaction design opportunities for music services in different contexts. These tests show the design research possibilities, benefits, and challenges of using a needfinding machine to explore context and its relation to user needs. In the process of "performing DJ Bot," the designers/wizards were able to explore people's connection with their music and potential needs that might drive future intelligent music recommendation agents and services.

5.1 Motivation

The DJ Bot project began as a way to test the ideas of the Needfinding Machine in relation to real-world interactive systems. The space of interactive music services is interesting because these services are heavily data-driven and powered by recommender systems (Schedl et al., 2015) but provide a product that is laden with personal meaning and contextual importance

(Hargreaves et al., 2002). Digital streaming music services allow people to access huge amounts of music and have changed the way listeners discover, share and curate their music collections (Lee and Price, 2016). The music recommendation systems behind these services can help listeners discover new music or suggest just the right song to play at the moment. In essence, these systems attempt to know the user to make predictions about what music they will enjoy (Rashid et al., 2002).

Music presents a rich and open test platform for our needfinding explorations. Everyone has both a biological and social connection to music (Trevarthen, 2002), allowing for almost anyone to be involved as a user in the development of new music interfaces. Music has been used for therapy to improve one's sense of purpose and is used a way to convey personal meaning to others (Amir, 1999; Bensimon and Gilboa, 2010), suggesting it as a useful mechanism for allowing designers to explore who the user is as a person. Nicholas Cook (2000) stated that deciding what music to listen to is a way of signaling who you are. Music is also linked to time and space and is used similarly to personal photo organization as a way of reminiscing and storytelling (Bentley et al., 2006) and creating a personal "musical panorama" of one's life (Frohne-Hagemann, 1998).

As people consume more music, interaction design around music listening is becoming more data-driven and focused on recommender systems. This is enabled by previously unseen patterns emerging from analyses of large data collections on listener behavior. For example, Zhang et al. (2013) analyzed Spotify listener data to determine when the most popular times during the day for listening were and what devices (mobile, desktop, web) were being used. In an analysis of six years of data from 310 Last.fm user profiles, Baur et al. (2012) were able to determine that seasons had a significant impact on listening habits. Still, vast stores of listener behavior data do not provide all the information makes people passionate about their music. While these studies highlight *how* users listen, they do not provide the richness of *why* users listen. Streaming services are now exploring alternative ways of categorizing music to get at this meaningful information. For example, Spotify's "Line-In" interface aims to collect more meaningful tags about music directly from users and plans to use this as meta-data in their recommendation system (Rottgers, 2018). Building on the desire for users to talk about their music, DJ Bot uses an interactive music agent as a platform to let the designer connect with a listener around music in-context. While situated in the listener's environment and performing as the machine, designers can explore new speech-based music service interactions and build their understanding of individual music listeners.

5.2 Pilot explorations of on-road music experiences

For our first context, we explore music listening while driving. The car provides a number of interesting opportunities for exploring the needs of music listeners, as it is a semi-private, semi-controlled environment where people often enjoy music or other audio-based media. From a logistical perspective, the car is readily instrumented with cameras, computers, and interactive devices. With the use of high-speed mobile routers, cars can be fully connected to the remote designer. Finally, music listening is one of the few safe secondary activities drivers can engage with. Current smart-phone based streaming services may be distracting or challenging to use, and present open design opportunities for new music applications.

5.2.1 Implementation

Functionally, DJ Bot in the car allows a designer to control a streaming music service on the listener's mobile device, communicate with the driver using real-time synthesized speech, and view multiple channels of live video and audio from the car. I built the DJ Bot prototype on top of the WoZ Way system (Martelaro and Ju, 2017) described in Chapter 4. I use the Spotify streaming service which allows for remote control using "Spotify Connect" from any device where the user is logged in. This allows the designers to use a desktop version of the application to control the music on the user's mobile device.

Figure 5.1 shows a system diagram outlining the remote designer locations, the communication streams, and the in-car interactions that occur with DJ Bot. Within the car, the listener connects their mobile device running the Spotify application to the car's audio system. Video cameras and microphones are placed around the car, allowing the remote designer to see both the driver and road from multiple angles. Having multiple views allows the designer to better experience the driving context. The road facing camera also helps the designer have a sense of the road conditions, allowing them to better plan interactions and avoid distracting the driver. A computer in the car streams live video and audio via a video chat client back to the remote designer. The computer also runs a text-to-speech engine and speaks messages sent from the remote designer through a separate portable speaker.

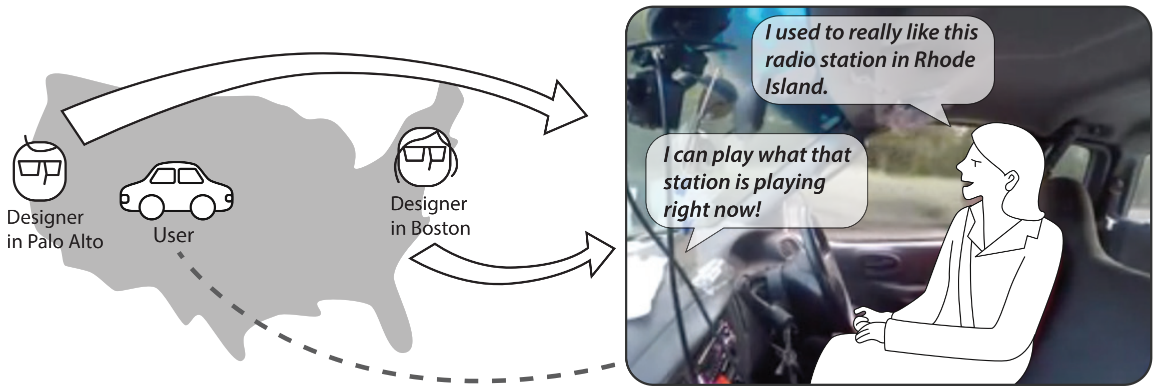


Figure 5.1: DJ Bot implementation for distributed design teams engaging with drivers. Designers can remotely interact with users from anywhere in the world, allowing situated, real-time needfinding through a machine.

The designer, acting as a wizard, controls DJ Bot through an interface that displays video from the car, the desktop Spotify application with "Spotify Connect" enabled, and a custom web-based interface for sending speech messages to the car, shown in Figure 5.2. The designer can view information such as the audio level and current song in the Spotify phone application and can control music using the application's audio player controls. The speech control interface includes a text input area to send custom messages and a list of scripted questions such as: "*What do you want to listen to next?*," "*Why did you choose that song?*," "*What does this song remind you of?*," and "*Can you tell me more?*"

To support documentation and analysis, video and audio are recorded using cameras mounted in the car. Because both sides of the interaction are required to reconstruct the dialogue, the designer's control interfaces are also recorded. All speech messages are logged and the music selections stored in the user's listening history.

5.2.2 Pilot Road Sessions

I conducted three interaction sessions for our initial exploration.

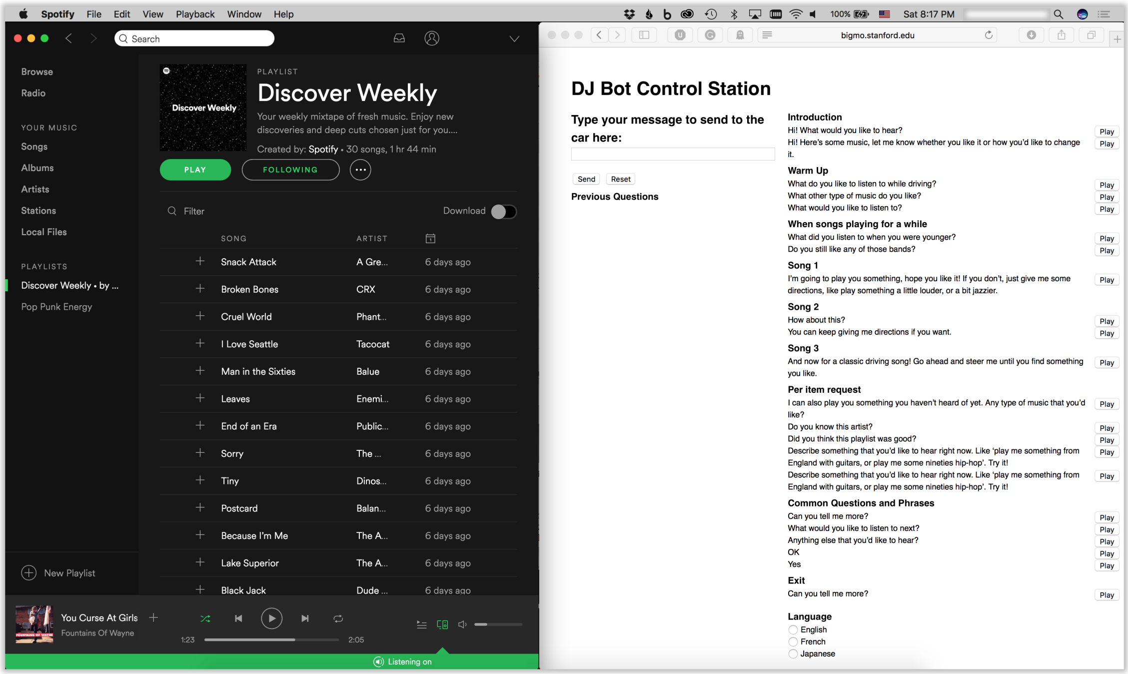


Figure 5.2: An example control interface from the DJ Bot pilot. The designer can remotely control music and synthesized speech with planned or improvised questions.

1. A one-hour session where I acted as the DJ/wizard for a user driving her pickup truck on the freeway

2. A one-hour session where a colleague and interaction design practitioner acted as a DJ/wizard for a user driving to visit a friend

3. A 30-minute session where two interaction researchers from Spotify acted as DJ/wizards for a commuter driving from home to school at 5:30 AM PST. One researcher connected from California while the other connected from Massachusetts, as shown in Figure 5.1.

After each session, I interviewed the wizards and drivers, asking questions about the interaction experience and the use of the Needfinding Machine. The goal in describing these sessions is to present a working example of how a needfinding machine can be used and to reflect on the opportunities for professional design work. The following sections provide excerpts and commentary from the sessions as a means to give the reader a feeling of the conversations that occurred between the designer performing DJ Bot and the listener/driver.

Pilot Session 1

During the first session, I controlled the DJ Bot for a user who drove her pickup truck on the freeway for about one hour. Before the session, I set up cameras and the data streaming computer in the user's truck. I had little prior information about the listener's musical preferences before starting the music. I began the session by asking "What would you like to listen to?" through the DJ Bot. This led to a back and forth interaction to determine if the listener wanted to hear the song that was selected. At one point in the interaction, the driver remembered a song that they liked and requested it to be played. I ended up playing an alternative remixed version of the song, leading to the following conversation:

♫ *I would walk 500 miles - remix* ♫

[DJ Bot] What do you like about this song?

[Driver] This isn't the original, is it? You said it was a remix. But it's just fun and it has a good beat and it's like easy to sing along to.

[DJ Bot] Do you sing along to music often?

[Driver] Yeah! It's pretty weird to sing along to music when I know I'm being recorded.

[DJ Bot] Do you sing along with others?

[Driver] Yes.

[DJ Bot] What songs you do you normally like to sing?

[Driver] Lots of songs. Like Disney songs.

This interaction elicited that the driver liked to sing along to music, enjoyed Disney music, and associated Disney music as music the user would sing with others.

While thinking of other music to listen to, the driver remembered a radio station they liked from a time when they lived in Rhode Island:

[Driver] I can give you a radio station in Rhode Island that I really like.

[DJ Bot] Okay.

[Driver] So it's 95.5 WBRU in Rhode Island. It's the Brown radio station, it's alternative rock. The stuff that they play is pretty great. When it's not overplayed songs.

[DJ Bot] Okay, let me see what I can find. Did you find this while you lived in Rhode Island?

[Driver] Correct. They introduced me to a lot of music.

[DJ Bot] Here's what they are playing now.

♫ *Green Day - Holiday* ♫

During the interaction, I switched windows from the control interface to a web browser and searched for the radio station's website. This quick thinking gave me a new source of potential music to recommend. It also presented an interesting example of the listener's relationship between place and music, providing future opportunities for further exploration.

While the system did function, this session was not entirely smooth, highlighting challenges for me and the user during the interaction. For example, I had trouble choosing songs since I was biased by my own musical tastes and did not have a baseline for what the driver liked. During the interview, I stated:

> Actually it's really challenging. I have really strong opinions about music and I kept seeing all these bands and was like ''I like this song! Let me play this song.'' But then I was trying to figure out ''How am I supposed to tease out of you what you like?'' And the thing is, I have such a limited knowledge of what I could even go off of.

Though this made the interaction more challenging, it pushed the driver to actively steer the music selection. This revealed information about the driver's preferences and kept them conversing rather than simply sitting back and listening.

The driver's comment about being recorded suggests that they were not always comfortable opening up to the machine. After the session, the driver stated:

> `Yeah, it was weird. It's also weird because I know I'm being recorded and I'm trying to drive, and I am telling someone who I don't know very well all about my musical tastes. Which is pretty intimate and so yeah, it was just weird on a lot of levels for me.`

This suggests some limitations of the Needfinding Machine method. In this particular case, the driver knew that a person was on the other end of the machine. Though this may have biased their answers during the session, the conversation afterward suggests even when the designer's intentions are known, eliciting and recording personal information may prove challenging. Still, much of the conversation went smoothly and provided ample information about the listener and their preferences. Moments of discomfort from the user may help designers to identify potential "off-limit" areas early in the design process. These points of tension also provide interesting moments to discuss with users.

Pilot Session 2

The second interaction session included Dr. Wendy Ju as a driver and a colleague who is an interaction design practitioner acting as the wizard. Wendy spent about 45 minutes driving one of our research vehicles on a freeway and scenic road on her way to visit a friend.

This session occurred after a recent and fierce presidential election. When the designer asked why Wendy chose a particular song, Wendy responded as follows:

> ♫ *Public Enemy - Public Enemy No. 1* ♫
>
> **[DJ Bot]** This one?
>
> **[Driver]** Yeah, I love this one.
>
> **[DJ Bot]** Why this right now?
>
> **[Driver]** Maybe like a post election thing and it has a lot of energy.

Although the question "Why this song?" was simple, it elicited a response with information about the song and about the context. In this case, the wizard built on the contextual information about the political times. This led to a more extended discussion of the relationship between current affairs and the driver's music.

After a while, however, Wendy was not interested in talking about politics and asked to change the subject.

[Driver] Let's not listen to any more Trump songs.

[DJ Bot] Okay.

[Driver] Like, let's play something by Missy Elliott.

[DJ Bot] How is this song?

♫ *Missy Elliot - Work It* ♫

[Driver] This is good.

[DJ Bot] Why Missy?

[Driver] I really like the way that she plays with words, I feel like she's like really really creative and like breaks rules. In like really interesting ways and, I don't know, it's like one of these things like, is so unlikely and then when you listen to it, it's kind of amazing and then it's very upbeat and I have difficulty understanding what's being said and I think it's like fun to figure it out, like a puzzle. And maybe a little bit like Devo, it just seems like she's having such a good time, such a good time singing, you know, and I love that.

Later in the session, Wendy spoke about bands from college and her connection with the artist *Fountains of Wayne*.

♫ *Fountains of Wayne* ♫

[DJ Bot] What do you like about this band?

> `[Driver]` `You know what, actually the thing I like about this band is it makes me think about college and it's a little bit funny because it's not something I actually listened to when I was in college, but when I went to grad school...`
>
> `...a lot of my thoughts about undergrad are colored by this soundtrack even though like I said I never listened to Fountains of Wayne in college. And I had like this homesickness for college...`

This then led to the wizard to ask "What other bands did you like in college?," which prompted Wendy to list off 14 other bands, helping to log a number of songs and genres that the driver enjoyed. This interaction showed the rich storytelling that can occur when thinking back on the music people enjoy. The story about college, in particular, paints a textured picture about the driver's life, helping the designer develop empathy for the driver and a sense of the meaning behind the 14 bands that were listed.

Pilot Session 3

The third session highlighted a number of strengths that a needfinding machine can have for remote needfinding. This session was conducted in a distributed manner with one designer at home in California and one at work in Massachusetts. To split the interaction load, one designer controlled DJ Bot's voice and one controlled the song selection. During this session, the designers communicated on a separate voice channel and coordinated their actions between music control and the bot voice. The session was done at 5:30 AM PST (8:30 AM EST), during the driver's 30-minute commute from their home in the city to school. While interacting through the machine, these researchers were able to experience the user's local context, despite the geographical, temporal and logistical challenges.[1]

Early in the session, the designers asked about what the driver listened to when they were younger.

> `[DJ Bot]` `What did you listen to when you were younger?`

[1] The driver's car was instrumented the evening before by the research team.

`[Driver]` Classical music. And a lot of Christian rock.

`[DJ Bot]` Do you still listen to that music anymore?

`[Driver]` I'm not really religious anymore.

After this comment from the driver, the remote designer controlling the DJ Bot voice moved on to another subject. However, after the session, she remarked that there was a tension in her own interest as a researcher and the role of performing the machine.

> When he said things for example about religion, I was like ''Oh!'' but then ''no, I probably shouldn't go'' you know the car goes digging around into your personal history. It wouldn't be on brand for the car or music service to go digging into your childhood.

This interaction further shows that even simple questions about one's music can lead to meaningful answers. However, in this case, the designers chose not to follow the topic. Being confronted with such a unique situation during conversation prompted the designers to reflect on how the machine ought to interact and what the machine should and should not talk about. The designers' in-the-moment and post-session reflection can be useful for understanding their own designer values and brings to light potential issues to consider for future design ideas.

5.3 Insights from DJ Bot Pilot Testing

5.3.1 Designing by, with and for data

By interacting through a needfinding machine, designers actively engage with and elicit data about people to understand their potential needs. When considering how designers should approach this data, Speed and Oberlander (2016) ask three questions around how we can design *by, with* and *for* data. Specifically, they consider:

1. How might designers develop new methods to capture data that reveals people's values in a respectful way?

2. How might designers capture how data influences people's behavior and design interventions that respect people's values?

3. How might designers mediate systems developed by other machines while considering people's values?

The Needfinding Machine is one method to address these three questions. By framing the interaction through the machine as needfinding, designers act and observe so that they can understand, empathize, and learn about the user's life and the user's values. Active interaction with the user, rather than covert surveillance of the user's behavior, allows the designer to explore useful data features while being sensitive to the user's values. During the first session with the driver who spoke about singing along to her music, the driver was acutely aware they were being recorded and interacting with a person through the machine. While the user's awareness may seem to inhibit needfinding, it engages the user in a participatory way, allowing them to better consider and control what they share with the designer. For example, in the second session with Wendy, we saw that Wendy would explicitly ask to change the subject of discussion. Though this cut off some avenues of conversation, it helped to guide the interaction in more relevant directions aligned with what Wendy would be comfortable discussing.

From the designer's perspective, we saw that by interacting through the machine, designers actively confront the implication of machines that elicit data from people. Interaction ideas and questions that feel okay in the abstract may turn out to be creepy or weird when implemented. During our pilot DJ Bot session with the driver on their early morning commute, the interaction researchers explicitly refrained from discussing the user's religion or childhood because they questioned if a machine *ought* to engage such discussions. The in-the-moment setup caused designers to consider what information can and should be used for the design of new music services. As the Internet of Things enables more data about users to be collected, designers will need to confront whether this data should be collected or used at all.

Instrumenting and documenting user interactions in context allows designers to see and understand how data flows through the context. The Needfinding Machine's functional elements allow for data to be captured and viewed in-the-moment and reviewed later during post-analysis. Capturing the data live allows designers to see how the information that is collected about the user is representative of the user's values. The live interaction allows designers to explore interventions that can enhance the user' experience and engage with the user's values. For example, the interaction during Wendy's drive indicated a political dimension to her music tastes. This, in turn, reveals aspects of Wendy's values to the designer. The designer can then work from this understanding of the user's values to assess what information is useful for the design. The designer can then consider how systems that collect data on their own or generate data, such as a music recommendation engine, might become better aligned with the user's values.

5.3.2 Understanding the person and the user

Bill Verplank argues that there are three key questions when designing interaction: *How do you feel?*, *How do you know?*, and *How do you do?* (Verplank, 2003). By conducting needfinding through a contextually situated system and by explicitly asking the user questions during the interaction, the designer can answer all these questions. The designer can ask how the user feels about the interaction and how the user knows what is happening during the interaction. The designer can also see what the user does during the interaction.

For example, during the second session, Wendy discussed a long list of bands she liked in college. This interaction helped the designers collect data about what music could be included in Wendy's listener profile. Additionally, the conversation allowed the designers to see how Wendy felt about the bands she listed and how she developed the feelings for the music. By getting the list of bands along with the personal meaning behind the bands, the designer could gather a set of meaningful information from the interaction. Information such as this could be directly used to design new features into a product, such as ways to seed new playlists or potential new voice commands. Seeing this meaningful information allows the designer to feel a connection to the interaction participant as a person rather than just another member of a user group. These data engage the designer with the user at a personal level, letting them see and probe the individual differences that they may have while using an interactive product.

The relationship between the designer and user does have some asymmetries due to the

Needfinding Machine setup. The interaction designers who participated in the third session during the morning commute noted that they felt that they learned a lot about the driver, whom they did not previously know, through the interaction. However, to the driver, the designers were still complete strangers. The interaction researchers stated "Oh, I should introduce myself!" during the post interview. This suggests that there are unresolved questions about how designers should frame these interactions, and how much reciprocity is expected from a needfinding machine. Should the user know that they are interacting with a designer? Should they know who that designer is? Moreover, how should the designer utilize the information gained to benefit the user? Interacting through the machine may give the designer an opportunity to reflect on these questions and on their own practices and values. Designers performing as the machine and eliciting meaningful information should consider how they want to engage with the user as they can understand both the functional aspects of the interaction and personal details about the user.

5.3.3 Implications of real-time interaction

Situated, real-time interaction supports designers in developing a rich view of the user's life in context. For example, one of the interaction researchers from the morning commute session noted that (virtually) being in the car at 5:30 AM was an eye-opening experience. The time of day painted a picture for the designer of an everyday user experience that they had not considered before. It was a departure from the designer's previous work with stationary voice interfaces and their experience as a remote wizard identified previously unknown needs around how people might listen to music as a means to wake up or ease into the day. The experience suggested that the interaction needs of the user might differ as the day goes on. This ultimately changed the interaction researcher's thinking about how often a music agent might interact based on the user's context.

Real-time interaction puts designers in an improvisational theater, where designers need to treat each utterance from the user as a gift to be responded to in kind (Johnstone, 2012). While planning is required for the logistics of the session, designers need to be very awake to the unplanned opportunities that open up in the course of an engagement. Reacting to events in-the-moment can give the designer the opportunity to understand experience right as it happens. Designers can improvise the machine's behavior as they are performing to explore different ideas quickly and to elicit different types of information. One readily improvised characteristic which can lead the designer to elicit different information is conversational style (Brennan, 1991). For example, the wizards interacting with the woman driving her pickup truck in the first session and with Wendy in the second session used more human-like conversation. This led them to focus on having deeper conversations about the music. During the morning commute session, the interaction researchers focused on a more machine-like interaction. Being more machine-like allowed them to shape the interaction to be closer to a what a product might be, but still allowed them to explore some of the more meaningful aspects of the user's music preferences.

We can liken the way interaction designers employ their intuitive and embodied sense of context and timing during in-the-moment dialogue to construct interaction with the way industrial designers and architects prototype and design in-the-moment with pliable materials. The industrial designer Henry Dreyfuss describes how using clay as a material allows the designer to explore form beyond what is possible with sketches (Dreyfuss, 1955). Working in three dimensions allows the designer to experience a model in a form closer to what the everyday experience would be. It also allows the designer to alter a design as they build, similar to how designers can alter as they sketch, but with less thought devoted to simulating what something may be like. The architect Eero Saarinen, for example, created "huge models that you could put your head into and really look around the architectural space and surfaces" (Rosen, 2016) as a means to experience the architectural design in one moment, and then rework them in the next.

The real-time interaction enabled by the Needfinding Machine parallels the designer's need for a tactile and embodied way to prototype a design *in situ*. Using a needfinding machine, the designer can get their head into the action and converse directly through the machine, shaping the interaction over time. The conversation with the user through the machine acts as the pliable material with which interaction designers can form new alternatives for future designs.

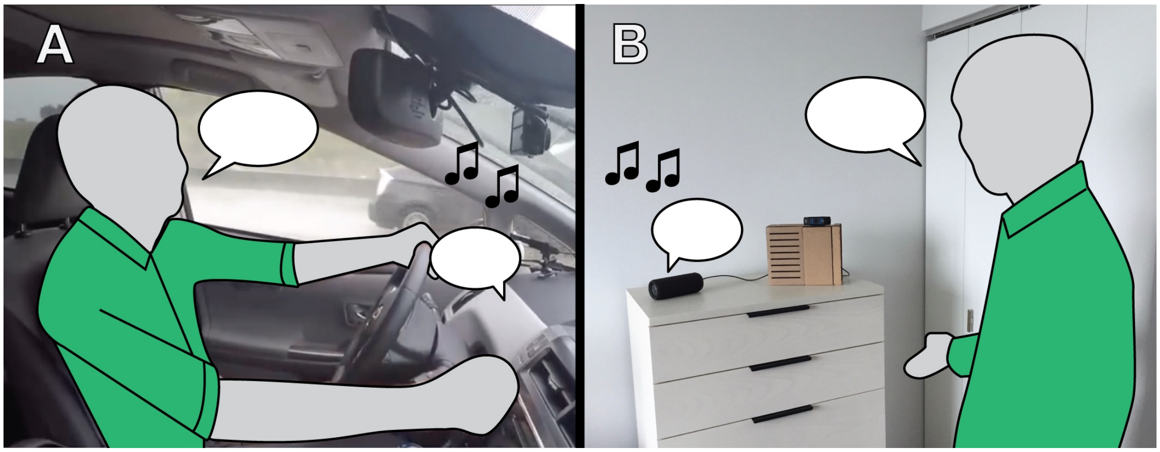


Figure 5.3: Sample images of exploration with DJ Bot in the a) car and b) home.

5.4 Exploring music interactions in context with Spotify

To better understand the feasibility of using a needfinding machine for user research in practice, I invited a team of interaction research practitioners at Spotify to conduct field deployments with DJ Bot in the car and the home. The team at Spotify is tasked with researching and developing insights around human-agent interaction in varying contexts.

5.4.1 Practitioners

Three members of the Spotify research team participated in five on-road Needfinding Machine sessions with different users and three in-home sessions with one user. The Spotify team members are experienced in academic research and industry-oriented user research. The three team members include:

1. **Henriette Cramer, Ph.D.** (Sr. Research Lead) - researcher focused on the dialogue between people and things that learn, adapt and personalize; the human side of machine learning. Henriette is particularly interested in the impact that design, data curation and metrics choices have on machine learning outcomes in recommendation, discovery and natural language settings. Location: San Francisco, CA

2. **Jenn Thom, Ph.D.** (Sr. Research Scientist) - researcher focused on how people ask questions in various socio-technical contexts, from speech interfaces to their online

social networks. Jenn uses qualitative and quantitative methods with deep experience in both consumer and enterprise products. Location: Boston, MA

3. **Sarah Mennicken, Ph.D.** (Research Scientist) - researcher focusing on the design of voice output and novel voice experiences. Sarah works at the intersection of user research, design, and technology helping to translate insights and designs into prototypes that allow studying tangible experiences. Location: San Francisco, CA

The three team members all have past experience and current needs around understanding how users can interact with near-future technologies. Their positions within Spotify allow them to provide useful and critical feedback about the Needfinding Machine and its use in a product-oriented team.

5.4.2 Exploring Music in Context

One goal of the Spotify team is to understand people's music listening experience. Spotify has 170 million monthly active users across 65 countries (Spotify, 2018). Log data alone cannot capture the richness of people's interaction with their music. Music is consumed in many places, often as a secondary activity (Greasley and Lamont, 2011; Juslin and Laukka, 2004; Juslin et al., 2008; North et al., 2004) throughout the day. People listen to music in their home, car, gym, and workplace on many different devices (Juslin and Laukka, 2004); the Spotify player is currently supported on desktops, TVs, voice assistants, cars, mobile phones, and gaming consoles (Spotify, 2018). Aspects of personal context, such as someone's current activity, emotions, motivations, and personal preferences also influences music listening behavior (Juslin and Laukka, 2004; Juslin et al., 2008; North et al., 2004; Greasley and Lamont, 2011). Music permeates so much of our lives and activities that researchers argue that listening behavior should be studied in context to understand the true nature of people's music experience (Juslin and Laukka, 2004; Juslin et al., 2008; North et al., 2004; Greasley and Lamont, 2011). Due to the context-dependence of music consumption, researchers interested in understanding user experience often employ in-person qualitative user research methods such as semi-structured interviews and contextual inquiry (Holtzblatt and Beyer, 2014). These methods provide a rich view of a person's life but can be challenging to accommodate in highly personal contexts such as the car or home. Researchers also use mobile-phone based experience sampling (Csikszentmihályi and Larson, 1987) to understand how people engage with their music throughout their everyday life (North et al., 2004; Juslin et al., 2008; Greasley and Lamont, 2011). This allows researchers to probe users in-context and in-the-moment. However, experience sampling requires participants to complete a questionnaire; this is not practical, for example, if people are cooking at home or driving their car to work. Experience sampling also requires the researcher to script what aspects of context to explore ahead of time. As I showed with WoZ Way, speech interaction can overcome these limitations: researchers can more naturally generate questions on-the-fly, and users can respond without disrupting their primary activity. Music interaction is also one of the most common uses for voice agent interaction (Bary, 2018) and extending voice control to conversation about music is now believable for everyday users.

5.4.3 DJ Bot Communication Structure

The Spotify team operated DJ Bot with two people, one music DJ and one speech agent controller. The Spotify team was geographically distributed between east and west coast offices. In addition, the team was often traveling and working from different offices or homes. These factors meant that the DJ Bot control interface needed to be accessible to all team members at the same time. Additionally, a communication channel needed to be set up between the remote team members so that they could talk with one another to coordinate their interaction. We call the interaction between DJ Bot and the participant the *front channel* and the communication between the remote researchers the *back channel*. Figure 5.4 shows the communication structure.

5.4.4 Methods for understanding the designer experience

I used two primary methods to understand how the interaction team used the Needfinding Machine within their design process, observations and semi-structured interviews. I conducted observations of the design team as they were engaging in an interaction session. I took notes during these sessions and reflected on them to facilitate the semi-structured

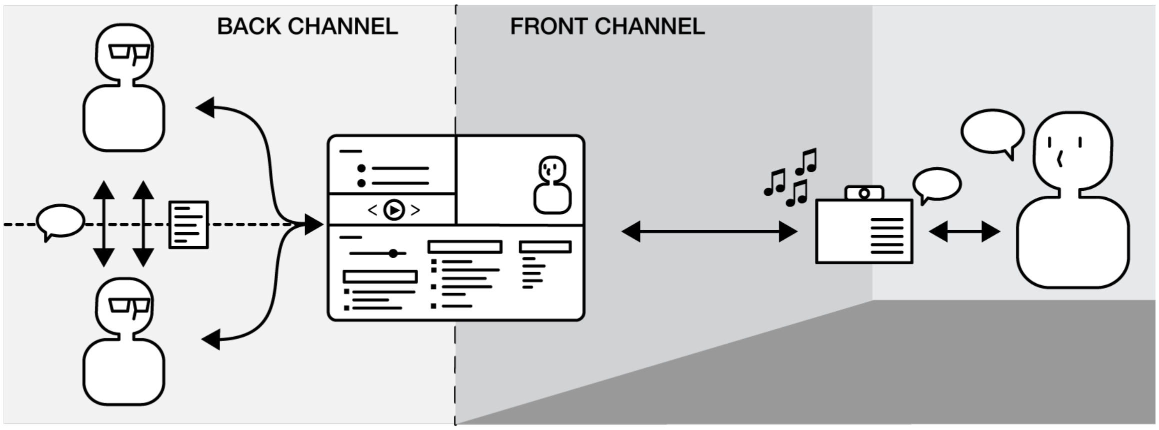


Figure 5.4: DJ Bot's communication channels. Front channel – user interaction with DJ Bot. Back channel – interaction with control interface, conversation between researchers via voice and text notes.

interviews with the design team. Short interviews were conducted after each interaction session that focused on the designers' experience during the session. One longer interview was conducted after all the on-road sessions. These interviews were focused on how the needfinding machine integrated and influenced the Spotify team's research process.

Observation

During each interaction session, I observed the design team members interacting with the users through the DJ Bot and with each other on a back channel. The observation included a live video feed from the user's location in the car or home and a live audio connection with the design team members. Video and audio were recorded during the session. Recordings were made locally (in-camera) in DJ Bot and remotely using a screen recorder. The interaction between the user and DJ bot was recorded on one channel while the back channel conversation between the designers was recorded on another audio channel. Since the designers are not speaking directly to the user through the machine but use text to speech, they can freely speak to each other. This allowed me to observe how the designers planned actions and discussed the interaction in real-time. Recorded sessions were transcribed to provide information to the design team and for the meta-analysis of the design interaction.

Interviews

We also used interviews to understand the experience of the design team while using the needfinding machine. Two types of interviews were conducted. Short interviews were done after each interaction session, and a longer interview conducted after all the driving sessions were complete.

The short post-session interviews were intended to capture information about the experience that the designers just had. Immediately after the user interaction session had completed, the designers interviewed with the user. This interview was done over video chat and was recorded. The designers focused on asking the user about the experience they just had with the DJ Bot. The designer's interview questions were aligned with the specific design questions that the team was looking to explore.

After the interview between the designers and user, I interviewed the designers about the session they had just completed. Our discussions focused on the experiences that the designers just had while using the DJ Bot as a needfinding machine. The goal for these interviews was to explore what went well during the session, what challenges there were, and what the team learned about their users.

I also conducted a longer interview with the Spotify team where they reflected on their overall experience using the Needfinding Machine across all the car sessions. These were semi-structured interviews which attempted to answer:

- What needs the team learned about their users?
- What did you learn about the system you are developing?
- What are you learning about the Needfinding Machine?
- What challenges have you had with the system?
- What opportunities do you see with the Needfinding Machine?

These questions were intended to get at how the designers experienced the Needfinding Machine and how it worked within their design process. Questions can be found in Appendix C. Though we did discuss specific learnings about the users in relation to the product that the team is developing, I will not focus the discussion here on this as this information

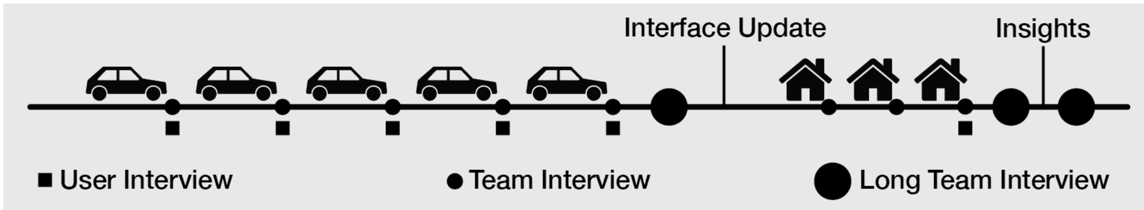


Figure 5.5: The DJ Bot research timeline. The Spotify team conducted five in-car studies with individual users and three in-home sessions with one user. I conducted two long interviews with the research team, then derived a set of insights with the whole team.

is private to the company. Instead, this section focuses on the meta-discussion about the Needfinding Machine and its use. These interviews were recorded and transcribed using an online transcription service.

The research timeline is shown in Figure 5.5. After each in-car session, the Spotify team conducted a short interview with the user about their interactions with DJ Bot. I then followed up the user interview with a short interview with the Spotify team focusing on their use of DJ Bot during the session. After all the in-car sessions were completed, I conducted a long interview the Spotify team's overall experience.

5.4.5 Research Timeline

Based on insights from the in-car sessions, I updated the DJ Bot control interface for use in the home. We conducted a set of pilot home studies to improve the control interface. We then engaged with one user for three interaction sessions over the course of one week. I conducted a short interview with the Spotify team after each interaction session. After the last in-home session, the Spotify team conducted a short user experience interview with the participant. I then conducted a long interview with the Spotify team focusing on the in-home sessions. I then led a second longer interview reflecting on both the in-car and in-home DJ Bot experiences. Finally, I reviewed the recorded sessions and interview transcripts and prepared a set of insights, benefits, limitations and best practices to present to the Spotify team. The entire research team then collaborated to reflect on and refine these insights.

5.5 Context 1: Music in the Car

The car is a common place where people enjoy music (Greasley and Lamont, 2011) and where listening is contingent on the driving activity. Music is also one of the few forms of entertainment that people can enjoy while driving. Since the driver can focus more on their music during a drive, it provides a prime environment for exploring how people think about and engage with their music in a specific context.

The car is also an environment where written methods of experience sampling cannot be used, but where people can have a conversation (Young et al., 2007). It should be noted that any secondary activity such as music listening or having a conversation can be distracting to the driver (Young et al., 2007). To mitigate risk during the driving sessions, the DJ Bot wizards were given video of the driver and road. They were also told to interact with drivers during less engaging moments of driving. Drivers were reminded to drive safely, avoid interacting with their phones manually and told that they could stop interacting with DJ Bot at any time.

Users

We invited five users to interact with DJ Bot in the Car. A description is provided for each driver including demographic information and the time of day which they participated. All users participated in the San Francisco Bay Area. These users were recruited from Stanford University. We aimed to recruit a diverse set of participants with varying music tastes and commute habits.

1. Male, mid 20's. Graduate student. Drove for 45 minutes on freeway. Afternoon.
2. Female, late 20's. Lab manager. Drove for 70 minutes from work to home. Afternoon.
3. Female, mid 30's. Lab manager. Drove for 45 minutes from work to home. Afternoon.
4. Female, early 30's. Administrator. Drove 45 minutes from work to home. Afternoon.
5. Male, late 60's. Self-employed. Drove for 45 minutes on freeway. Afternoon.

Figure 5.6: A video frame of a driver interacting with DJ Bot in the car.

5.5.1 Driving Session Description

The remote design team interacted with drivers using DJ Bot in the car during all five drives. Each drive was conducted in either the driver's personal vehicle or in a lab vehicle that was set up for data collection. Three drivers participated in their own vehicles, and two drivers participated in our research vehicle. Users drove for approximately 30 – 70 minutes. The three drivers who drove in their own vehicles drove on their everyday commutes from work to home. The two drivers who participated in the research vehicle drove back and forth along a scenic highway during the middle of the day.

For sessions with drivers participating in their own vehicle, a video and data collection system was set up within their car approximately 30 minutes before they departed. A sample video frame of the driver is shown in Figure 5.6 Once the video recording and DJ Bot system were set up, an experimenter then had participants review and sign an informed consent form. The experimenter then allowed the driver to seat themselves in the vehicle. Once seated, the experimenter helped the driver setup their phone to run the Spotify mobile application and set the phone to never go to sleep, ensuring that the application maintained an internet connection for remote control. The experimenter then explained that the driver would only need to interact with the system using speech and that they would not need to interact with their phone using touch. The driver was told first and foremost to drive safely. Beyond that, they were allowed to behave as they normally would as they drive. The experimenter then turned on all the cameras and asked DJ Bot to say "hello," signaling to the remote designers that the driver was ready to depart. Once the DJ Bot said "hello," the experimenter allowed the driver to ask any final questions about the study and then allowed them to depart.

During the drive, the remote practitioners controlled the music and interacted with the driver using the DJ Bot text-to-speech system. The remote designers observed the live camera feed from the road and asked the driver questions. The team used a simple script they created before the drive to explore different in-car interactions. The utterances explored different potential interactions and included introductions and greetings, sample songs, questions about what to play, common questions, and follow-up statements. The remote designers could also use the free text area to write custom statements or questions. During each session, the designers loosely followed the script while leaving room for free interaction based on what the driver was interested in.

5.5.2 Interaction Description

In this section, we describe what happened during the five sessions. The first part of this section reviews the more quantitative aspects of the interaction dynamics, including the frequency of interaction and the amount of time spent speaking across the session. The second subsection gives a more qualitative picture of the sessions and describes information gleaned from the post-interviews with users.

Interaction Dynamics

Across the five sessions, the DJ Bot made an average of $M = 70.6, SD = 25.2$ utterances while Drivers made an average of $M = 80.8, SD = 33.0$. DJ Bot spoke for an average of $M = 135, SD = 35$ seconds while Drivers spoke for an average of $M = 290, SD = 165$ seconds. When normalized for time spent driving, DJ Bot spent about $M = 4.1\%, SD = .6\%$ of the time speaking, while Drivers spent on average, $M = 8\%, SD = 4\%$ of the time speaking. Figure 5.7 compares the percent of time spent speaking during the drive of both

DJ Bot and the Driver for each of the five interaction sessions. Overall, the interaction researcher speaking through DJ Bot spent approximately the same amount of the drive speaking through DJ Bot and spent less than 5% speaking. Session four was the only session where the interaction researchers spoke about half as much in other sessions. After this session, the researchers said that they spent less time speaking because the driver's chosen songs were long, and they decided to not interact during songs as the driver seemed to prefer not speaking while the music was playing.

In general, drives spoke for more of the drive than DJ Bot, however drivers 2 and 4 spoke for less than 5% of the drive. These drivers treated the interaction in a more transactional manner, mostly making requests for songs to be played, similar to how currently available speech agents interact with users. Drivers 1, 3 and 5 spent more time speaking during their drive, speaking 12.6%, 7.8%, and 12.1%, respectively. During these sessions, drivers often spent more time discussing their music choices and answering questions from DJ Bot.

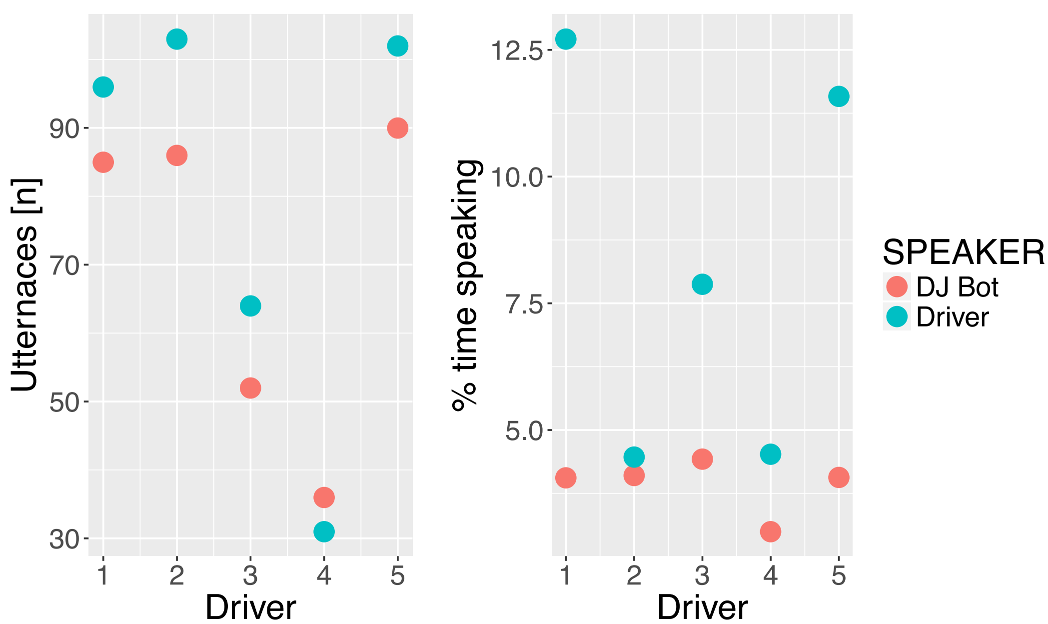


Figure 5.7: Plots of the interaction dynamics during the DJ Bot in the car sessions including number of speech utterances and percentage of time speaking during each drive.

5.5.3 Qualitative interaction

During the sessions, the interaction research team engaged the drivers in various ways while using DJ Bot. This section is organized by types of interactions that occurred during the drive.

Requesting Music

Drivers took different approaches to requesting music. The interaction researchers prompted drivers to try different ways of interacting with DJ Bot. The Spotify team scripted a few statements that would suggest ways for the driver to request music. As part of their interaction prototyping, the team was interested in exploring music requests beyond explicit requests for a specific artist or song. To explore more varied music requests, the team included the following prompts in their script:

1. *I'm going to play you something, hope you like it. If you don't, just give me some directions, like, play something a little louder or a bit jazzier.*
2. *You can keep giving me directions if you want.*
3. *And now for a classic driving song! Go ahead and steer me until you find something you like.*
4. *I can also play you something you haven't heard of yet. Any type of music you'd like to hear?*
5. *Describe something you'd like to hear right now. Like 'play me something from England with guitars' or 'play me some 90's hip-hop'*
6. *What would you like to listen to next?*
7. *Anything else you want to hear?*
8. *What would you like to listen to?*

Exploring new music

During the driving sessions, the interaction team also prompted users to explore new music. This was done by suggesting songs that were not in the user's playlists or were discussed during earlier parts of the interaction. Driver 1 explored many new tracks and engaged with the wizards in trying to find new music to listen to. During the post-interaction interview, the Spotify team asked the driver how their experience during this session was different from the last time that the driver had driven. The driver responded to this question by discussing exploring new music:

> **[Driver]** Oh, there was just much more new things, right? I would not, while I'm driving I don't have the capacity to go out and look through Spotify for something. So, I think it was really nice to actually get suggestions and listen to other things that I would never have listened to just by myself.

This suggests that conversational music discovery could be a potential use of voice interaction in the car. This type of interaction does not currently exist, however, our sessions provide some exploration of a possible way to engage listeners with new music exploration.

Driver 1 continued to discuss his experience discovering new music during other parts of the interview. When asked about what he wished the system could have done, the driver said that he would have liked more time with the experience. The research team then followed up with asking what he would do if he had more time:

> **[Driver]** I would try to get more different impressions, like more different music, just to listen to, just to learn about. Because, while driving, I don't really have anything else to do so it's a great way to get to know more things or catch up on these bands that I have never ... I would just not go out and just look for it, but now I have the time and the tool to do that. So, that's neat.

Again, this suggests a possible design opportunity for allowing drivers to engage more with their music in a setting that allows them to focus on the music. Furthermore, from a need perspective, this suggests that the driver does not always have much time or the mindset to discover new music and learn about new music. There is a need to branch out, but there is something that might hold people back in the other parts of their lives.

Exploring new music was also a way for the research team to explore when people did not like music and how to react and respond. During Driver 3's interaction, the team learned that the driver used to listen to more hip-hop, but had since changed their preferences to Christian music. Though the team tried to suggest hip-hop that the driver had not heard, the driver was often not pleased with the music choices. The researchers said that it made song selection challenging because they did not know if they were acceptable to play, but they wanted to see what would happen if they did. The specificity of the driver's requests ended up pushing the researchers into a smaller area of music than in previous sessions.

> **[Researcher]** You get a slightly weird trade-off with what should we do now? What's the right thing to do? Which, we didn't have with the other participants because they didn't seem to have such a specific wish, and it was more of, let's play some old-time music, they liked it, and it felt a bit with her that we ended up in this little box of us being comfortable with playing particular things for her.

This session suggests an opportunity for how a system will need to respond to making selections that the listener isn't happy with. If a music speech system intends to suggest new music proactively, then considerations should be taken for how to deal with making poor suggestions. Interacting through a needfinding machine can help the design team both understand the range of poor song requests, explore the listener's rationale behind why the song request was poor and explore possible repair strategies.

Talking about music and memories

During our pilot sessions, the research team found that asking what people listened to when they were younger often led to people talking in depth about their music history.

Furthermore, a recent analysis of Spotify data by Stephens-Davidowitz (2018) suggests that people's musical tastes often align with the music they listened to in their teen years. Drivers typically responded with more lengthy answers to this question, with the response duration being on average 13.7 seconds long, compared to the average response length of 2.8 seconds long for all speech utterances. Of all the things that people spoke about, any comments made in relation to some memory were very rich with both personal information and information about the driver's music preferences. Below are some responses to the question *"what did you listen to when you were younger?"*

[Driver 1] `Oh, I was listening to like, System Of A Down and Linkin Park, and I still sometimes do but I don't know, Linkin Park has become really kitschy, and I'm not a teenager anymore, you know. (laughing)`

[Driver 2] `Um ... Boyz II Men. I also listened to some Aaliyah.`

[Driver 3] `Um, a lot of 90's R&B. Um, mainly a lot of 90's R&B. And then, when I moved to California, I got into the whole Bay Area rap, but I don't listen to it anymore. I used to listen to alternative too, when I was in high school. I grew out of that.`

[Driver 4] `When I was younger, I listened to a lot of 80's music, like Phil Collins, uh, Journey, Survivor, Foreigner, Tears for Fears, um, Sheila-E, Prince, uhhm, El Debarge.`

[Driver 5] `Like those uh, pop music and uh, '80s and '90s.`

While drivers typically spoke about personal memories while talking about music they listened to when they were younger, drivers would also comment on other memories with some music connection. For example, Driver 1 talked about the music he played in a band when "The Passenger" by Iggy Pop came on:

[DJ Bot] `This is Iggy Pop, The Passenger.`

[Driver 1] Oh, really? I love this song, I actually played it in a band. I love it.

[DJ Bot] What else did you play?

[Driver 1] I played the drums. Oh, ''what else did you play?'', that's what you're asking. Do you know from Golden Earring, Radar Love? That's a neat one, also fits nice to that. A Fool For Your Stockings, ZZ Top. Billy Idol, a couple of songs from him like White Wedding. I don't know, I can't really think of them right now, but there we're a couple.

During this interaction, the song that was played spurred a memory and the driver disclosed their relationship to the music. The driver also listed many other songs that he played in a band. While playing music in a band may not be as universal as associating music with your youth, there may be design opportunities for connecting music with memories as a way to build playlists or preference models based on conversation.

In some cases, listeners also associated music with people in their lives as they think back on their musical memories. Driver 5 discusses some of the musical tastes that he has developed based on the musical preferences of his son:

[Driver 5] Actually, I haven't been listening to songs for quite a while.

[DJ Bot] Good to get you listening again.

[Driver 5] Yeah.

[DJ Bot] Welcome back.

[Driver 5] Thanks. Most I listen to is my son's playing... like Muse, he plays all the time, so those are the ones that I listen in the last few years. I don't know, when people get older, whether they are into the pop songs anymore.

[DJ Bot] Can you tell me more?

> **[Driver 5]** Well when I was young, I listen to every song that is on the radio. But, starting maybe about 10 years ago, I stop listen to the radio. And that's how I don't listen to song anymore. And the only way I can, get in touch with other song is from my son. He listens to like Japanese songs, and some of these like Muse, and some pop music, much more... I would say less, uh... noisy song. I'm not sure... I'm not sure whether other older people did that or not.

This interaction gives a more individual picture of how this driver's music tastes have changed over the years. The driver in this interaction was also older than the other participants and their conversations around music shed light on how people remember older music and find newer music as they age, as they develop different social relationships, and as technologies change.

Thinking about ones musical history can also be an enjoyable activity. We found that many of the participants enjoyed talking about their music and their past. The following quote from Driver 2 during the post-interview suggests that she enjoyed the experience of thinking about the music she used to listen to when she was younger. One thing to note here is that when asked specifically about what she liked to listen to, Driver 2 only listed two artists. Still, this question appeared to have primed her to think back on her past in order to guide DJ Bot in playing songs.

> **[Driver]** Also, this was sort of fun. My commute was over an hour long, but it almost felt like a game. Like, talking to tech, I guess, and trying to remember songs that I listened to back in the day. She would have conversations with me, too. She'd initiate questions like, what did I used to listen to when I was younger? What type of genres I tend to like? And she made jokes which was awesome. I thought the jokes were great.

Within the car context, this driver's feedback suggests that conversation around music might be a fun way to reduce the perceived time of a long commute. Though we only ran one session with each driver and it is possible that we may have some risk of a novelty effect, it may be worth future exploration of how to structure self-reflection around music as a way to improve commutes and help people engage more with their music service.

Overall, memories about music formed much of the conversation around musical choices. The interactions presented in the sessions suggest that people may have a need to reminisce about their music. They also present potential opportunities within the design space for using memory as a way to engage the listener in conversation.

5.6 Context 2: Music in the Home

The interaction research team at Spotify was also interested in exploring music interaction in the context of the home. Studying users in their homes would provide a different view of a user's life and would show the range of contexts that DJ Bot could work. The goal of DJ Bot in the home is to allow the practitioners to explore interaction dynamics with a user who is mobile within the home, with groups of people listening to one device, and during times when users have many entertainment options beyond audio-only systems.

The focus of this exploration was to see how DJ Bot could be used in the home and how using a needfinding machine in the home differed from the car. The researchers were also interested in exploring multiple engagements with a single user. For this part of the case study, the research team recruited one participant to interact with DJ Bot three times over the course of one week. In this section, I describe the DJ Bot in the home system and the three sessions that the research team had with the participant.

5.6.1 DJ Bot in the Home System

DJ Bot in the home was designed as a single device that connects to a user's audio system. This is similar to the Amazon Echo Dot, where the intelligence and speech capabilities reside on the device (and the connection to the internet), and the audio is played through the user's speaker system of their preference. The physical DJ Bot is shown in Figure 5.3b and is a stylized cardboard box reminiscent of simple mid-modern radios. This simple aesthetic was chosen so the visual design would not be overpowering or indicative of a certain product personality. Cardboard is used to indicate to the user that they are using a prototype and not a fully finished product. A webcam is placed on top of the device, clearly visible to the user. The camera, a Logitech C920, includes a blue light that is visible to the user so that they know when the camera is recording and when the camera is off. In addition, users are provided with a cover for the webcam to give them even more peace of mind when they are not participating in an interaction session. Audio is sent to the user's speaker system via a 3.5 mm audio cable. If users do not provide their own speaker system, a USB powered speaker is provided along with the DJ Bot. The speakers are placed at least 3 feet away from the camera so that the music sound level does not overpower the webcam microphone and make it challenging to hear the user. DJ Bot in the home is powered by an Apple Mac Mini mounted vertically inside of the cardboard box. The Mac Mini runs both the streaming music application and the speech interaction application. With both applications running on the same machine, the music and speech audio both play on the single audio output and through the speaker system. This allows for DJ Bot's voice to be collocated with the music. The speech application was modified to control the audio levels of the music streaming application. Whenever a message is sent from the control software, the speech application lowers the music volume by half. The system then restores the volume after DJ Bot finishes speaking. The designer also has manual control of the music volume.

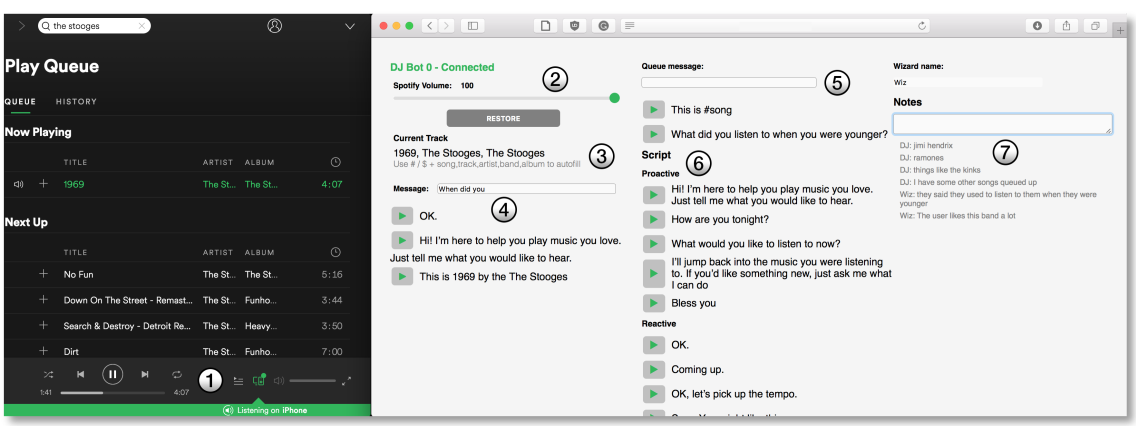


Figure 5.8: DJ Bot in the Home control interface. (1) Music control through Spotify desktop application, (2) Volume control, (3) Live song data, (4) Custom message input, (5) Queued message input, (6) Scripted speech messages, (7) Collaborative notes.

Remote Control Interface

The remote control interface was updated based on feedback from the DJ Bot in the car sessions. Shown in Figure 5.8, the interface still includes the free message text input and the prepared questions areas. The third column is provided for the designers to take notes. These notes are not spoken out loud by the DJ Bot but are recorded in the DJ Bot log file. This synchronizes the designer's notes with the local video recording made on the DJ Bot, making it possible to review the notes while reviewing the moment during the interaction that the note was written.

Data Logging

As with the DJ Bot in the Car sessions, video and audio were recorded locally and remotely from the video stream. Local video recording was accomplished using Apple QuickTime webcam recording. The remote recording was accomplished by capturing the screen on a remote computer with the video chat set to full screen.

The designer back channel was also recorded remotely on a separate audio channel from the DJ Bot interaction front channel. This channel was audio-only due to screen recording restrictions within Spotify.

All messages from the remote designers to the DJ Bot were timestamped and recorded in a log file. This included messages for DJ Bot to speak, audio level changes, and notes. Data from the song was also recorded. This allowed the designers to have a list of the music that was played during the session. The metadata of the song included the artist, album, length, and Spotify catalog data. All data were logged using ISO 8601 UTC timestamps.

5.6.2 Procedure

One member of the research team visited the user's home for 30 minutes and set up DJ Bot. The researcher explained that the goal of the study was to explore in-home interaction with music and had the participant sign an informed consent form. The participant was told that the remote team would log in to DJ Bot and have access to the camera and the audio stream. With this in mind, the participant chose a place to put DJ Bot. In this case, the user decided

Figure 5.9: A sample video frame of the remote view from DJ Bot in the home.

to place DJ Bot in their bedroom looking across the room at a window. A sample frame of the remote video from the DJ Bot's perspective is shown in Figure 5.9.

The team planned three 90-minute interaction sessions. The team emailed the user a day in advance of each session with questions for the user or tips on using DJ Bot. These included suggestions on how to request relaxing or energizing music before the second session and asked about upcoming activities before the third session. An hour before each session, the team emailed a reminder to the participant. The team then logged into DJ Bot, turned on the camera, started data recording and interacted with the user for an hour and a half. After the session, the team stopped the recording systems, turned off the camera, and sent an email confirming the conclusion of the session.

5.6.3 Interaction Dynamics

The home sessions were different in terms of interaction dynamics compared to the in-car sessions. During each session, DJ Bot spoke 31, 21, and 13 times respectively. Overall, the user treated their interaction with DJ Bot transactionally, requesting or skipping songs. Occasionally, the user asked for facts about an artist. The user did try out different ways of requesting music based on the email prompts. However, there was little conversation around the music. The team noted down what activities the user seemed to be doing. The user did many other activities during the session, such as cleaning their room, folding laundry, taking phone calls, and talking with other housemates.

5.7 Insights from the interaction research team from Spotify

We refined the learnings from our team interviews and developed a set four insight themes: 1) context exploration of the environment and target audience, 2) being a speech agent, 3) reflection, and 4) flexibility and improvisation.

5.7.1 Context exploration of the environment and user

The Spotify team discussed how using a speech agent in an everyday environment allowed for rapid exploration of the contextual factors that influence music listening. One researcher noted, "*By being remote and by having the synthesized voice, that is very useful to let the user keep the illusion of their real context. I think that is really good. Maybe that is why it is so strong to figuring out new contexts or new setups.*"

While DJ Bot arguably introduces another actor into the context, the team perceived this interaction to be less disruptive than if they were present in the user's home. It reduced the influence an in-person experimenter would have, and, as a result, reactions were more natural. For example, during the in-car sessions, users would sing along to their favorite songs or yell at other drivers as if they were alone. During the home session, the user would audibly sigh about poorly chosen songs but did not mention their displeasure about those same songs during the post-interview.

The researchers noted how they were able to alternate focus between the environment, the situation, and the user easily without the participant noticing. Changing focus from the user could be considered anti-social or distracting in person. The DJ Bot set up gave the team more flexibility to change focus as needed, broadening their observational range. The contextual insights helped the interaction researchers to focus and refine their post-session interview protocols.

Another theme that emerged was that, through the combination of simultaneous conversation and contextual observations, the researchers could quickly get to know a user. One researcher noted, "*We could see the participants, and we felt like we knew so much about them*," even before the post-session interview.

DJ Bot provided information that the team could not typically capture during an in-person interaction. For example, live video from multiple angles in the car provides a unique view of a driver's behavior with their music and the road. The team perceived this to improve their understanding of the environmental context of driving. However, for the home, the fixed-view video did not prove as useful during the home session since the user was often out of frame. This affected the team's ability to gain a broader sense of the overall home context but still allowed them to gain insight into music listening as a background activity.

The Spotify team also noted that being in the actual context and reducing their influence on the user meant that interaction could be sparse. Acting as a speech agent allowed the researchers to drop into the background for long stretches of time. The realistic interaction frequency gives researchers opportunities to explore other aspects of context that they would not otherwise be able to see in-person or through experience sampling.

5.7.2 Being an embedded speech agent

The Spotify team felt that their interactions through DJ Bot were not that different from having a conversation or doing contextual inquiry — just that the conversation was mediated through a robot's voice. On the other hand, the team would have liked to benefit more from the capabilities of currently available voice assistants. For example, while the team could provide more intelligent responses through DJ Bot, they felt that the time it took to type out what the DJ Bot spoke stilted the interaction. In the context of music, it is also mostly impossible to match the speed and personalization of automated song selection based on sophisticated recommender systems. Occasional network breakdowns also prevented the researchers from interacting with the participant. Interestingly, this seemed to be more of a concern to the practitioners than for the participants and made the team realize that any real-world system should be able to handle network breakdowns or issues interpreting the user. While some users noticed slow response time or poor song choices, most attributed it to the realities of voice technology. Many users felt that the interaction was a value-add, noting that it was enjoyable to talk about music with a bot.

The team also discussed how they worked to act more “bot-like” during the sessions to avoid seeming too smart or making their presence apparent which could potentially break the context. For example, the researchers would discuss via the back channel what interactions would fit within the situation. In some cases, users did feel that the speech agent guise was broken, though it was not apparent that this changed people’s behavior much during the session.

5.7.3 Reflection throughout the research process

A recurring theme that came up in our interviews was how interacting through DJ Bot allowed the team to reflect on the interaction sessions in-the-moment and throughout their research process. The team heavily used the audio back channel to discuss what they were seeing and learning about the user and context collaboratively. The team also used the back channel to plan their next actions based on the current situation. This allowed them to conduct very short cycles of questioning and testing followed by discussing what they saw, what they might do next, and whether their approach had to be adjusted to yield better insights. One researcher said, *“we could have this conversation about ‘Are we getting this out of this? Yes? No?”’* This opportunity to directly interact with the participant while being able to follow up with each other is like a think-aloud study focusing on their process of conversing with the user. For example, during the first in-home session, the team had developed an interaction plan, but halfway through, realized that their plan was not working out as expected. The team discussed what to do and formed a new plan which they used for the rest of the session. This planning is similar to lab-based user tests, where in-person researchers talk with the observing team ‘behind the glass’ during breaks or after a session. By interacting through a speech agent, the research team can change course without interrupting or waiting until after a session.

The researchers also commented on how useful the collaborative notes were while interacting through DJ Bot. The notes created another communication channel where the researchers could comment on important information about the user and capture their thoughts about the interaction. Although the team was geographically distributed, the participating team members could see the other researchers' thoughts and follow up with them on the voice back channel. All members of the team could influence the interaction as the speech agent represented everyone, changing the dynamic from more tradition user research where one team member acts on the team's behalf.

Since all the notes are sent to the DJ Bot and time synchronized with the DJ Bot speech interactions, song information, and video, the team could quickly review the notes and plan for future sessions. These session logs reminded the researchers of what happened during the session without needing to review all the video recordings.

5.7.4 Interaction Flexibility

DJ Bot allowed the interaction research team to explore more open-ended ideas than they would if they were testing with an automated system or asking users questions via a survey. By acting as the speech agent, the team could work improvisationally, asking the participants questions based on the situation at hand and following up on the user's responses and requests immediately. A researcher stated, "*You can have it say whatever you want. You can have more exploratory type conversations - surprising types of conversations.*" The team commented positively about the additional range of interactions they could have with the user: "*Depending on how exploratory would want to be, you can make it more or less structured.*" This theme came out when considering the different sessions. The team spoke about how the early in-car sessions were open-ended and constructive. The team found that after having developed a plan and many scripted speech interactions, they could focus their inquiry while still being flexible. One researcher likened this to jazz improvisation, where there is an underlying structure but also the freedom to alter their interaction based on the situation.

Interacting remotely also allowed the team to collaborate in a way they had never been able to do before. One researcher noted "*It is, also, good for teams that, like us, are geographically not co-located. That has been a tremendous difference because if we had a*

Wizard of Oz set up that would just work in Boston or San Francisco, that would be much harder. This way, it made it really easy to be in three different places and still all be in it."

Finally, working remotely through the speech agent allowed the team to assume different roles during the interaction sessions. The researchers would take turns acting as the music controller or the speech agent, without the user being distracted by the change. Acting through the speech agent offered further flexibility in what the team focused on throughout an interaction session, whether they wanted to spend time observing the situation, conversing with the user, talking with their teammates or planning the next set of songs. The researchers considered this to be a larger advantage over in-person interaction as any changes to their focus or roles during a session might seem odd to the participant.

5.8 Conclusion

The DJ Bot sessions echoed some insights from the WoZ Way interaction sessions in Chapter 4. The designers using DJ Bot could experience the user's context and ask the user questions while remaining behind the scenes. By being remote, the team could explore both the user's environment and their internal motivations in a way that was similar to being in person, but ultimately allowed the team more flexibility in what they focused on during the session. Just as one of the researchers in the WoZ Way deployment discussed being able to sit back and observe the environment, the Spotify researchers felt that they could shift their focus from the user to the context without disturbing the user. In addition to this, the DJ Bot sessions show how *teams* interacting with each other on a back channel allow for conversation in real-time during a needfinding machine interaction. This provided a benefit beyond the WoZ Way sessions as the back channel conversations allowed the team to reflect in-the-moment. These reflective conversations allowed the team to plan their interaction collaboratively, generate insights about the session more quickly, and also allowed them to challenge their thinking about how the interaction with DJ Bot ought to be. Overall, the DJ Bot sessions solidify insights from the earlier studies in this thesis and present new benefits of designers interacting through a needfinding machine.

Chapter 6

Conclusion

As interactive devices change the nature of many products and create whole new classes of devices, basing design on user needs is more important than ever. This work begins with the idea that finding and understanding unmet user needs is the foundation of useful, usable, and meaningful product design. To understand these needs, designers must connect with the end-users of a product. As physical products become imbued with computation and networking capabilities, designers can begin to use information collected from products in a manner similar to how designers collect and use information from web-based products. At the same time, designers can also use these same computational capabilities to facilitate a mediated conversation with the user, allowing them to stay in-the-loop and leverage their observational and creative abilities when finding user needs.

This dissertation has explored how designers can understand the needs of people in the context of interactive devices. The primary goal of this work is to provide designers with a new means of needfinding. The Needfinding Machine extends designer's perceptual reach and allows them to understand user needs in relation to the diverse and rich contexts of people's everyday experiences. The Needfinding Machine does not intend to replace the human design researcher but aims to augment their abilities.

6.1 Contributions

The primary contribution of this dissertation is the proof-of-concept design and evaluation of the Needfinding Machine, a method for conducting needfinding by having designers interact with users *through* interactive objects in the real world. The Needfinding Machine augments and amplifies a designer's ability to explore user experience remotely. Designers using a needfinding machine can situate themselves in the user's context while leveraging the capabilities of an interactive device's information gathering features.

This dissertation shows a design tool that acts along the continuum of design work. The Needfinding Machine can be used for early-stage needfinding, iterative prototyping, and design development, and has the potential to be used after a product has shipped to provide continual feedback to a design team.

The specific case studies in this dissertation show how a needfinding machine can be used in the contexts of human-robot interaction with students, driver-car interaction with advanced driving assistance, and human-agent interaction with a music-oriented conversational agent. The implementations and evaluations within these three areas accomplish the following research goals:

1. Verification of if and how people will interact with robots via conversation about their interactions with a product and their personal concerns

2. Implementation of a technical framework to enable remote interaction prototyping and observation with interactive devices embedded in the real world

3. Validation of the benefits and limitations of using Needfinding Machines in professional design work

Interacting through the Needfinding Machine provides designers with a new way of doing needfinding and enables a form of working that has not been previously possible. As with any design method, there are different benefits and limitations to using the Needfinding Machine that designers should consider.

6.2 Benefits

Being remotely embedded within a user's everyday context allows practitioners to explore the user's environment with more flexibility than in-person interactions. By acting through a needfinding machine, which is assumed to be a part of the context, the interviewer's presence can be less biasing or disruptive for the participant. Compared to in-person sessions, remote researchers gain flexibility in their observational range, as they can shift their focus to the user, the environment, or the data throughout an interaction without distracting the participant. As working through a needfinding machine also inserts a layer between the user-product interaction and the researcher-researcher interaction, it allows researchers to verbally collaborate with each other to plan and reflect on the session in-the-moment without bothering the user. Such conversational in-the-moment reflection also allows the researchers to generate insights effectively as the researchers interact with the user. Using a needfinding machine enables globally distributed team members to participate in the process of understanding their users together. Finally, we found that interacting through a needfinding machine allows practitioners to explore a range of interaction from open-ended conversation to focused questioning and interaction prototyping. With a semi-structured plan, a trained design team can rapidly explore new contexts and interaction opportunities through improvisation with users directly in real-world environments.

6.3 Limitations

Although I have discussed many benefits that a needfinding machine can have for interaction designers, I have also identified some limitations in their usage. During the interaction sessions, some people were uncomfortable with being recorded. Often this was an initial discomfort that passed as the interaction continued. In some cases, users did not want to engage beyond a certain point during the interaction. For example, drivers in the Renault sessions would stop talking or reduce their engagement in the conversation based on the road condition. Or, the driver talking about politics in a DJ Bot session asked specifically to change subjects. While these moments reduce what a designer can learn from conversation, they also are useful as feedback. Moments where users close themselves off may highlight important aspects of the environment or personal context that a designer should consider in their future products.

There can be issues on the designer's end when considering what information they have been given and how they should proceed with the conversation. The designer may question how they should act as an interested researcher versus how they should perform the machine, potentially muddling their needfinding efforts. In the Renault sessions, wizards would sometimes question when it was even appropriate to speak to the driver based on what was happening on the road. During a DJ Bot session, the designers moved away from a line of questioning that prompted the user to discuss their past experience with religion. Though the designers stopped their questioning, these moments acted as points of reflection for the designers. The moments where a designer is confronted with information that conflicts them could potentially help the designer consider their own values during the design process (Sengers et al., 2005).

Understanding a new context through a needfinding machine can be limited by how much the remote team can see and hear in the user's environment. For example, having only audio reduces what a researcher can understand about the physical situation. There is significant planning required to ensure that the right data streams are provided to the remote researcher. This also requires significant coordination and specialized audio/video equipment to view and record a channel in the user's environment and provide a back channel for the research team. Unlike experience-sampling, these tests can generate a large amount of semi-structured data that can be challenging to quickly derive insights from. This can increase the time required to analyze the data, making it less practical for practitioners in industry working on a fast-paced team. This being said, even without reviewing the recorded sessions, the in-the-moment learnings are still useful for a research team.

From a systems perspective, it is tempting to fall into the trap of adding "bells and whistles" that enable higher fidelity prototyping and realism. Instead, it is important to develop the system so that it maintains focus on the actual needs of the user (Mantei, 1986; Grudin et al., 1987). For this, I recommend that the design of a needfinding machine interface evolve over numerous sessions.

There are also limits to what environments and products can be instrumented. Spaces such as the home and office, with ubiquitous WiFi and sufficient power, can easily accommodate for the functional components of a needfinding machine. Even our case study in the car was relatively easy to instrument. At present, the Needfinding Machine depends upon having environments with easy network access, power, and the ability to host cameras, microphones, and the interactive system itself. Adaptations to remove these types of requirements will enable us to better perform needfinding in less-resourced environments, where better longitudinal needfinding is direly needed. Developments in embedded computing and global network connectivity, as well as careful considerations of bandwidth needs, might open needfinding up to these new arenas.

While human operators can improvise intelligently, it can be challenging to recreate what a data-driven or automated system would be able to do. These inconsistencies can break the experience of interacting with an interactive device. Being more emerged in the actual context and being able to respond to the situation at hand can also make more controlled testing challenging. Even getting through a script of interactions was challenging when the user or context demanded the interaction move in a different direction.

Finally, the team from Spotify noted that documenting and sharing the data from Needfinding Machine sessions is challenging within the corporate IT environment. Aside from the technical knowledge required to build a needfinding machine, instrumenting and recording the live interaction is beyond what many designers can easily perform today. More consideration must also be given to how best to communicate the results of the multidimensional data and in-the-moment learning that is collected from the user, system logs, and designer reflections so that they are easily shareable and interpretable, and can more meaningfully guide product discussions.

6.4 Considerations for Using Needfinding Machines

By going through multiple iterations of Needfinding Machine prototypes, several study sessions in different contexts and different researcher team setups, I along with the practitioners I worked with have identified several best practices for using Needfinding Machines in remote context explorations.

- *Consider and respect user privacy.* By respecting privacy, the designer can engender more trust and elicit more honest feedback. Users who do not feel spied on are more likely to engage in conversation with the interactive device. To respect user privacy, consider only observing and interacting during specified times. Give participants a reasonable notification ahead of time to inform them about the modality and the duration of an active session. Frame the engagement as a conversation not as a covert observation. Make users feel comfortable stopping their interaction at any time.

- *Frame the conversation as a needfinding exercise.* The Needfinding Machine is intended to extend a designer's perceptual reach and needfinding capabilities through conversation. By being a good conversation partner and by framing the conversation around the user and their experience, the designer is more likely to elicit information that is relevant for their design process. Consider best practices for conducting contextual inquiry and user interviews. Avoid selling a design to the user. Let the user show you and describe to you their honest experience.

- *Have a human designer or design team in-the-loop.* The Needfinding Machine is specifically designed as a human-in-the-loop method so that the designer can gain the most benefit from observing and interacting with the user. The human designer uses their intelligence to understand the user's speech, interpret the user's behaviors, and question the user based on the situated context. While it may be possible to automate some forms of questioning, current machine-based natural language and speech recognition technologies cannot handle the complex and rich information and interaction done in a needfinding situation. Furthermore, the designer ultimately is the creative agent that will develop new design ideas to satisfy a user need.

- *Identify a preliminary set of rules on how to interact with the study participants to follow during an interaction session.* By making a preliminary plan, the designer can guide an interaction to feel more or less machine-like. For example, should you react to context changes you can only detect as a human observer? Will you proactively talk to the participant after a specific duration of time? Consider what aspects of the context you are interested in looking at and make sure that your data collection systems capture this. Make sure that your plans are flexible and be open to improvising in-the-moment based on what you see in context.

- *Setup a back channel for interacting with your collaborators during the session and test it ahead of time.* This could be voice or text-based. Take time to reflect on the experience in-the-moment and afterward. Evaluate the outcome against the goals of the session.

- *Start with minimal features in a control interface and iteratively develop over time.* You might discover additional functionality or automation as you go along. Taking notes on what turns out to be useful or requires changes can make development more focused and reveals insights about what information or control is important for the research team and the future product.

- *Synchronize data streams and notes with the captured video.* Since in-context sessions could be without a time limit, they can lead to large amounts of data. Properly timestamped data will allow for rapid post analysis and lets you find and focus on the moments of interaction.

6.5 Closing

From the studies in this dissertation, I have found that using needfinding machines can be a useful way of doing problem space exploration in-context with real users. By remotely interacting through a needfinding machine, practitioners can do design research in ways that are similar to methods such as contextual inquiry, but with the flexibility to work remotely and to reflect with collaborators in-the-moment. While the presented case studies explored understanding people's interactions with tutor robots, advanced driving systems, and music players, I imagine that other practitioners developing products which are context-sensitive, such at home appliances, personal robots, or even health devices, can benefit from doing real-world exploration through a needfinding machine.

The implementations that I presented in this thesis focused primarily on the design of interactive physical products. While the Needfinding Machine is concerned with designing interaction for products, we can also imagine developing needfinding machines for social service design. For example, there may be opportunities for health service designers to develop and use a needfinding machine for the design of in-home health interventions. As designers use more computing technology and automation to augment their services, using a needfinding machine becomes more feasible and may provide insights to service designer beyond more traditional service design methods.

As interactive devices become more integrated into people's lives, they could also be used for continual understanding of user experience. There are opportunities for future research on how to scale their observations and interactions. For example, developing interfaces that allow small teams to understand the experience of many people or developing protocols that allow practitioners to check in on real users while respecting user's privacy. We might also consider developing methods for semi-automating the questions we ask users about their experience so that designers can extend their reach to more users and focus more on unique or highly engaging moments. Overall, interactive devices can provide a new communication link for designers to explore the contextually situated aspects of people's experience with technology in their everyday lives.

While addressing user needs will always be the primary goal of product designers, the types of products and the methods we use to develop them are continually evolving. As products become more computationally enabled and more intelligent in their interactions with the user, designers need to understand how these new capabilities can be used to satisfy user needs. With intelligence comes a focus on interaction between users and products. However, while computing systems allow for products to be smarter, in the words of Terry Winograd, "Successful interaction design requires a shift from seeing the machinery to seeing the lives of the people using it" (Winograd, 1997). More than ever, we need design methods that can help designers connect with and understand their users so that we can develop technologies that are aligned with the social values we wish to uphold. I believe that designers will play a large role in defining how intelligent systems and services interact with people. The Needfinding Machine presents one vision for a future where designers can use interactive systems to see the lives of people using new technology.

Remember our opening story of the designer working on smart kitchen appliances. This designer intended to create "smart" kitchen appliances that would delight their users. All the technology is in place to create an intelligent and interactive fridge or coffee maker. However, the technology alone cannot help the designer to create a great product and to develop a set of interactions that provide value and delight to the user. By employing the

Needfinding Machine, the designer can connect with the user through prototypes of their smart kitchen appliances so that they can engage with the user in their everyday kitchen context. The Needfinding Machine allows the designer to see the lives of people by seeing *through* the machinery. By interacting with the user in context and through a needfinding machine, the designer can begin to explore the questions of how their appliances might behave towards and communicate with the user. Furthermore, the designer can explore what needs the user may actually have based on their real-world experience. Exploring the design space through a needfinding machine is as much about what *not* to make as what to make. We may find that the designer has determined that many of the "smart" capabilities that they had initially thought of are not as useful or desirable by people. However, their interactions through a needfinding machine may reveal a hidden need that sparks an entirely new design idea. I hope that with increased perceptual reach, designers can better see the lives of people and create products that make for more human-centered and value-driven interactions with everyday products.

Appendix A

The Interaction Engine

In this appendix[1], I give a brief history of smart product development and discuss the technology that now enables designers to create needfinding machines. I describe the system architecture and underlying components based on widely available computing hardware that I have developed for building my own needfinding machines. I call this system the Interaction Engine. The Interaction Engine is an enabling architecture for thinking about coordinating communication between networked physical hardware. The Interaction Engine description provides both a high level outline of the system features required for creating interactive needfinding machines as well as a detailed schematic of the technical systems used within the interactive devices I have created for this thesis.

A.1 Interactive Devices

We are entering a golden age for interactive device design. Concepts that have long existed as mere visions, like Vannevar Bush's Memex (Bush, 1945), Marc Weiser's Tabs, Pads and Boards (Weiser, 1995), and Apple's Knowledge Navigator (Sculley, 1987) are now a commercial reality. High-end products such as the Nest thermostat, Drop Cam security camera, Amazon Echo interactive speaker, and Phillips Hue light bulb are inspiring designers to contemplate how products can change by adding networked interactive hardware. What is

[1]This appendix is adapted from parts of Nikolas Martelaro, Wendy Ju and Mark Horowitz. 2018. The Interaction Engine. *In Design Thinking Research*, pp. 147-169. Springer, Cham, 2018.

most exciting is that the basic components that make such concepts possible are accessible to savvy designers, not just seasoned engineers (Hartmann and Wright, 2013). Just as the interaction design community's adoption of embedded microcontrollers has driven a breakthrough in physical computing (Banzi and Shiloh, 2014), enabling artists and designers "more creative thought and further iterations" on their designs (Gibb, 2010), the adoption of lightweight and networked computing platforms can power a similar revolution of multimodal connected devices. Various efforts have begun to introduce these technologies to the interactive device design community (Berdahl and Ju, 2011; Kubitza et al., 2013; Overholt and Møbius, 2013; Vandevelde et al., 2015; Martelaro et al., 2016b). The emerging design pattern uses small networked computers, such as the Raspberry Pi or BeagleBone, with embedded microcontrollers, PC peripherals, open-source software, and connection to cloud-computing services to enable designers to more easily explore a wide variety of novel, connected, post-PC devices (Kuniavsky, 2010). These components allow for designer to use video, audio, data processing, wireless communication, and physical actuation on relatively low-cost devices. These multimedia functionalities are the same functionalities that are used in needfinding machines. Video, audio, and data can be streamed between the designer and user, providing the communication link for allowing real-time conversation. The designer can also take advantage of the networked capabilities of the hardware to remotely control different interfaces in the user's environment. In the next section, I describe the primary requirements for creating a needfinding machine based on the Interaction Engine.

A.2 The Interaction Engine

The name Interaction Engine hearkens back to the early 20th century, when a revolution in inexpensive and widely available motor technologies enabled a wide variety of new products from the automobile to the home kitchen stand mixer. Cheap and widely available embedded and networked computing systems are now the new engines that enable a wide variety of interactive devices, from voice assistants to smart home security systems. We can think of an Interaction Engine as a set of computational tools and capabilities that we can embed into products in order to enable both interactivity and a link back to the designer. One can think of this framework as an embeddable composite material for new device design (Vallgårda

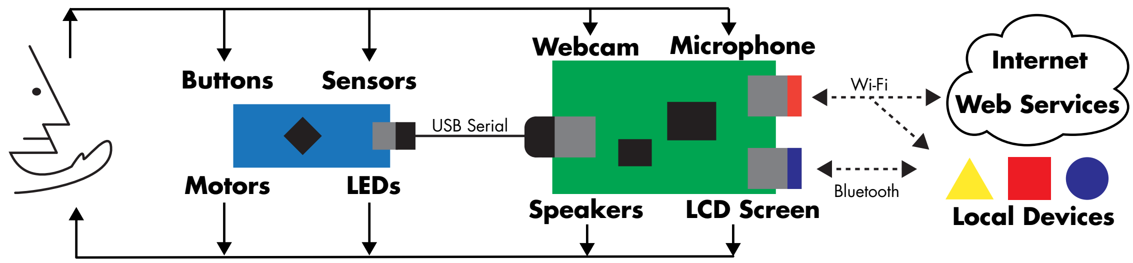


Figure A.1: The Interaction Engine – an architecture including the basic components of an internet connected, interactive device.

and Redström, 2007) consisting of a general purpose computing system, and open-source software, and connection to networked computing services.

Operating system-based computer

An operating-system-based computer serves as the foundational computing unit of the Interaction Engine. In the past, these systems were full laptop or small form-factor PC's, however a new class of single-board computers now allow designers to build the computers directly into the devices. These systems are small, inexpensive and typically run some flavor of the Linux operating system. One popular system is the Raspberry Pi, a $35 USD credit card sized computer built around the ARM processor; however, for specific applications, other systems with more or less power may be preferred. These single-board computers (SBC) are updated on a regular basis; it makes sense to size the specific board to the complexity of the computation in each application. The provision of an operating system provides commonality in a diverse and rapidly changing market, while giving designers the ability to choose systems based on computing power, peripherals, and form factor, without much switching cost. Using an operating system also means that we have access to a wide array of off-the-shelf software to quickly and easily enable different input, output, data processing, and communication features. Using an operating system also allows designers to quickly add features to their prototypes without significant redesign of their hardware.

Physical Computing Interface

A microcontroller unit (MCU) often serves as an interface to the immediate, physical world. Although most many embedded computers feature General Purpose IO ports (GPIO) which can act as the physical computing interface, these interfaces are not standardized. Often, separate MCUs are used for standardized physical computing (Russell, 2012; Palazzetti, 2015). Although different MCUs have widely varying peripherals such as ADCs, PWM, and various communication interfaces, one critical MCU feature within an Interaction Engine is some form of connectivity and communication, such as serial over USB, with the upstream OS-based computer. The ability to decouple the physical interface of the MCU from the processing capability of the SBC is valuable for modularity, extensibility, and upgradeability. This benefits the design process as designers can develop stable physical interfaces and interchange them with different SBCs allowing them to use SBCs that are best for their interactions. In my own projects, I have often added features that require greater computational power or different input/output capabilities in the middle of the project and can simply plug our physical computing interface into a new computer and port the high-level software in little time.

Connection to the cloud

Network connectivity is a requirement for building a needfinding machine. Near ubiquitous WiFi and cellular service allows for devices to always be online. This allows designer to remotely monitor and control needfinding machines embedded in users' everyday contexts. In order to enable a connection to the internet, any computing systems will require some networking interface such as Ethernet, WiFi, Bluetooth, or cellular. Many of these interfaces are already built into most OS based computers or can be added with little cost and development time.

The network interface also allows the Interaction Engine to gather data from the cloud, outsource high-computation tasks, store information remotely, and interact with Web services such as Twitter, Spotify, Amazon AWS, or Internet of Things platforms. This connectivity effectively embeds these remote services as material inside an object (Vallgårda and Sokoler, 2010). For designers, this is one of the most enabling aspects of the Interaction Engine as the

OS-based computer allows them the use of mature and robust interfaces for interacting with the cloud. This provides designers with a huge area of exploration for embodied, tangible interaction with digital services and data by extending potential designs beyond the physical and local world.

Appendix B

TutorBot Study 2 Questionnaire

TutorBot Questionnaire

Start of Block: Researcher Input Before Participant Begins Survey

R01 (Researcher ONLY) Please enter participant ID:

____

R02 (Researcher ONLY) Confirm participant ID:

____

R03 (Researcher ONLY) Please enter your initials:

____

R04 Vulnerability Manipulation

○ Vulnerable (1)

○ Not Vulnerable (2)

R05 Expressivity Manipulation

○ Expressive (1)

○ Not Expressive (2)

End of Block: Researcher Input Before Participant Begins Survey

Start of Block: Instructions

I01 **Thank you** for signing up to participate in our study today.
After our pre-activity survey, please proceed on to watch a brief video about the activity.
Once complete, the study leader will guide you to the next room to begin the activity.
If at any point in the process you want to stop, please vocalize out loud for your study leader to respond and assist you.

End of Block: Instructions

Start of Block: Demographics

01 Please indicate your gender:

- ○ Male (1)
- ○ Female (2)
- ○ Prefer not to answer (3)

---

02 Please check the label(s) that would describe your race and/or ethnicity:

- ☐ Hispanic/ Latino (1)
- ☐ American Indian/ Alaska Native (2)
- ☐ Asian (3)
- ☐ Black/ African American (4)
- ☐ Native Hawaiian/ Other Pacific Islander (5)
- ☐ White (6)
- ☐ Prefer not to answer (7)

---

03 How old are you?

- ◯ 14 (1)
- ◯ 15 (2)
- ◯ 16 (3)
- ◯ 17 (4)
- ◯ 18 (5)

End of Block: Demographics

Start of Block: Personality - Pre Activity

04 How much would you agree or disagree with the following descriptions of your personality?

	Strongly disagree (1)	(2)	(3)	(4)	(5)	(6)	Strongly agree (7)
Reserved, quiet (1)	○	○	○	○	○	○	○
Open to new experiences, complex (2)	○	○	○	○	○	○	○
Extroverted, enthusiastic (4)	○	○	○	○	○	○	○
Critical, quarrelsome (5)	○	○	○	○	○	○	○
Conventional, uncreative (8)	○	○	○	○	○	○	○
Sympathetic, warm (7)	○	○	○	○	○	○	○
Disorganized, careless (10)	○	○	○	○	○	○	○
Calm, emotionally stable (11)	○	○	○	○	○	○	○
Dependable, self-disciplined (12)	○	○	○	○	○	○	○
Anxious, easily upset (13)	○	○	○	○	○	○	○

End of Block: Personality - Pre Activity

Start of Block: Experience with Interactive tech - Pre Activity

I02 During today's task you will be learning how to work with electronics and computer programming. We do not expect you to have any experience with these, but if you do we would like to know.

05
How frequently or infrequently do you interact with...

	Never (1)	(2)	(3)	(4)	(5)	(6)	All of the time (7)
Siri (18)	○	○	○	○	○	○	○
Automated phone operators (19)	○	○	○	○	○	○	○
Similar responsive audiovisual technologies (20)	○	○	○	○	○	○	○
Facebook wall posts of pictures, updates (21)	○	○	○	○	○	○	○
Public Twitter tweets (22)	○	○	○	○	○	○	○
Similar social networks to a broad spectrum of acquaintances (23)	○	○	○	○	○	○	○

End of Block: Experience with Interactive tech - Pre Activity

---

Start of Block: Experience - Pre Activity

06 How much or little experience do you have with each of the following activities?

	No experience (1)	(2)	(3)	(4)	(5)	(6)	A lot of experience (7)
Building things? (1)	○	○	○	○	○	○	○
Writing a short computer program? (3)	○	○	○	○	○	○	○
Soldering two electronics components together? (4)	○	○	○	○	○	○	○
Drawing a diagram of a simple circuit (e.g. a light with a battery)? (5)	○	○	○	○	○	○	○
Making a simple circuit (e.g. a light and a battery)? (6)	○	○	○	○	○	○	○
Figuring out why a simple circuit isn’t working. (2)	○	○	○	○	○	○	○
Learning how to make complex electronic devices? (7)	○	○	○	○	○	○	○

I03 There are a few concepts today's activity will cover. Below, please give your best guess of how to define each key term.

---

07 What is a resistor?

- ○ A component that helps to manage the flow of current so it does not damage other electronic components. (1)
- ○ A component that spins when current flows through it and creates a rotating magnetic field that spins tiny magnets connected to its shaft. (2)
- ○ A component that is like a "brain" for circuits that can use code that you have written to do mathematical functions. (3)
- ○ A component used to store charge temporarily, consisting of two metallic plates insulated from each other by a dielectric. (4)

---

08 What is a microprocessor?

- ○ A component that helps to manage the flow of current so it does not damage other electronic components. (1)
- ○ A component that spins when current flows through it and creates a rotating magnetic field that spins tiny magnets connected to its shaft. (2)
- ○ A component that is like a "brain" for circuits that can use code that you have written to do mathematical functions. (3)
- ○ A component used to store charge temporarily, consisting of two metallic plates insulated from each other by a dielectric. (4)

---

09 What is a motor?

- ○ A component that helps to manage the flow of current so it does not damage other electronic components. (1)
- ○ A component that spins when current flows through it and creates a rotating magnetic field that spins tiny magnets connected to its shaft. (2)
- ○ A component that is like a "brain" for circuits that can use code that you have written to do mathematical functions. (3)
- ○ A component used to store charge temporarily, consisting of two metallic plates insulated from each other by a dielectric. (4)

End of Block: Experience - Pre Activity

Start of Block: Video - Pre Activity

I04

Page Break

I05
- PLEASE STOP HERE -

Your study leader will welcome you into the activity room now.

Page Break

End of Block: Video - Pre Activity

Start of Block: Engagement - Post Activity

I06 The following questions will ask you about your experience during the tutorial.
If you have not completed the tutorial, please do not click the next page.

10 Please answer the following questions based on today's LED circuit building exercise.

	Not at all (1)	(2)	(3)	(4)	(5)	(6)	Very much (7)
If I could build circuits everyday, I would look forward to it. (2)	○	○	○	○	○	○	○
When I was building the circuit, I felt bursting with energy. (3)	○	○	○	○	○	○	○
I persevered even when things did not go well in my circuitry. (4)	○	○	○	○	○	○	○
I could continue doing this sort of activity for hours. (5)	○	○	○	○	○	○	○
I was very resilient, mentally. (6)	○	○	○	○	○	○	○
I felt strong and vigorous when I was building the circuit. (7)	○	○	○	○	○	○	○
To me, building my circuit was challenging. (8)	○	○	○	○	○	○	○

My work on the circuit inspired me. (9)							
I was enthusiastic about circuit building. (10)							
I was proud of the circuit I built today. (11)							
I found the work full of meaning and purpose. (12)							
When I was working, I forgot everything else around me. (13)							
Time flew when I was working. (14)							
I got carried away when working. (15)							
It was difficult to detach myself from my work. (16)							
I was immersed in my work. (17)							

I felt happy while working intensely. (18)	○	○	○	○	○	○	○

End of Block: Engagement - Post Activity

Start of Block: Trust - Post Activity

11 Please recall the robot that was talking you through the exercise of building the blinking LED circuit. How well or poorly do the following phrases describe the robot?

	Describes very poorly (1)	(2)	(3)	(4)	(5)	(6)	Describes very well (7)
The robot exhibited technical competence. (1)	○	○	○	○	○	○	○
The robot was interested in my well-being, not just its own. (6)	○	○	○	○	○	○	○
The robot was virtuous. (4)	○	○	○	○	○	○	○
If I needed help, the robot would do its best to help. (5)	○	○	○	○	○	○	○
The robot was truthful. (7)	○	○	○	○	○	○	○
The robot exhibited professional behavior. (2)	○	○	○	○	○	○	○
The robot displayed a warm and caring attitude. (8)	○	○	○	○	○	○	○
The robot passed on new information or ideas that were helpful. (9)	○	○	○	○	○	○	○
The robot listened carefully. (10)	○	○	○	○	○	○	○
The robot was honest. (3)	○	○	○	○	○	○	○

End of Block: Trust - Post Activity

Start of Block: Disclosure - Post Activity

12 Please respond to the statements below based on your experience during today's task:

	Strongly disagree (1)	(2)	(3)	(4)	(5)	(6)	Strongly agree (7)
I did not talk about myself. (2)	○	○	○	○	○	○	○
My statements of my feelings were usually brief. (15)	○	○	○	○	○	○	○
I spoke about myself for fairly long periods of time. (14)	○	○	○	○	○	○	○
My conversations were short when discussing myself. (18)	○	○	○	○	○	○	○
I talked about myself. (3)	○	○	○	○	○	○	○
I discussed feelings about myself. (19)	○	○	○	○	○	○	○
I disclosed who I really am, openly. (20)	○	○	○	○	○	○	○
Once I got started, my disclosures lasted a long time. (21)	○	○	○	○	○	○	○
I revealed information about myself without intending to. (22)	○	○	○	○	○	○	○
I sometimes did not control my disclosure of personal things I said. (23)	○	○	○	○	○	○	○
Once I got started, I fully revealed myself in my disclosures. (24)	○	○	○	○	○	○	○

End of Block: Disclosure - Post Activity

Start of Block: Social Presence - Post Activity

13 Please answer the following questions about the robot you worked with today.

How much or little...

	Not at all (1)	(2)	(3)	(4)	(5)	(6)	Very much (7)
...did you feel as if you were interacting with an intelligent being? (1)	○	○	○	○	○	○	○
...did you feel as if you were alone? (2)	○	○	○	○	○	○	○
...did you feel involved with the robot? (3)	○	○	○	○	○	○	○
...did you feel as if you were in the company of an intelligent being when interacting with the robot? (4)	○	○	○	○	○	○	○
...attention did you pay to the robot? (5)	○	○	○	○	○	○	○
...did you feel as if the robot was responding to you? (6)	○	○	○	○	○	○	○
...did you feel as if you and the robot were communicating with each other? (7)	○	○	○	○	○	○	○
...did you feel as if you and the robot were a team? (8)	○	○	○	○	○	○	○
...did you feel that the robot wanted you to succeed? (9)	○	○	○	○	○	○	○

End of Block: Social Presence - Post Activity

Start of Block: Social Facilitation

22 Please answer the following questions based on today's task.

	Not at all (1)	(2)	(3)	(4)	(5)	(6)	Very much (7)
I felt like I had a companion during this task. (1)	○	○	○	○	○	○	○
I was entirely alone during this task. (2)	○	○	○	○	○	○	○
This was a solitary task. (3)	○	○	○	○	○	○	○
This was a shared activity. (4)	○	○	○	○	○	○	○
There were social aspects to the task. (5)	○	○	○	○	○	○	○

End of Block: Social Facilitation

Start of Block: Companionship - Post Activity

14 During you interaction with the robot how much or little did you feel that the robot was...

	Not at all (1)	(2)	(3)	(4)	(5)	(6)	Very much (7)
Good (1)	○	○	○	○	○	○	○
Loving (2)	○	○	○	○	○	○	○
Friendly (3)	○	○	○	○	○	○	○
Cuddly (4)	○	○	○	○	○	○	○
Warm (5)	○	○	○	○	○	○	○
Pleasant (6)	○	○	○	○	○	○	○
Kind (7)	○	○	○	○	○	○	○
Sweet (8)	○	○	○	○	○	○	○
Close (9)	○	○	○	○	○	○	○

End of Block: Companionship - Post Activity

Start of Block: Perception of the robot

19 During the interaction, the robot was most like a...

- ○ peer (1)
- ○ tutor (2)
- ○ teacher (3)
- ○ boss (4)

End of Block: Perception of the robot

Start of Block: Learning - Post Activity

I06 There are a few concepts that today's activity covered. Below, please give your best guess of how to define each key term.

---

15 What is a microprocessor?

- ○ A component that helps to manage the flow of current so it does not damage other electronic components. (1)
- ○ A component that spins when current flows through it and creates a rotating magnetic field that spins tiny magnets connected to its shaft. (2)
- ○ A component that is like a "brain" for circuits that can use code that you have written to do mathematical functions. (3)
- ○ A component used to store charge temporarily, consisting of two metallic plates insulated from each other by a dielectric. (4)

---

16 What is a resistor?

○ A component that helps to manage the flow of current so it does not damage other electronic components. (1)

○ A component that spins when current flows through it and creates a rotating magnetic field that spins tiny magnets connected to its shaft. (2)

○ A component that is like a "brain" for circuits that can use code that you have written to do mathematical functions. (3)

○ A component used to store charge temporarily, consisting of two metallic plates insulated from each other by a dielectric. (4)

17 What is a motor?

○ A component that helps to manage the flow of current so it does not damage other electronic components. (1)

○ A component that spins when current flows through it and creates a rotating magnetic field that spins tiny magnets connected to its shaft. (2)

○ A component that is like a "brain" for circuits that can use code that you have written to do mathematical functions. (3)

○ A component used to store charge temporarily, consisting of two metallic plates insulated from each other by a dielectric. (4)

18 Please indicate how much you agree or disagree with the following statements.

	Strongly disagree (30)	(31)	(32)	(33)	(34)	(35)	Strongly agree (36)
I am confident that I can write a short computer program. (1)	○	○	○	○	○	○	○
I am confident that I can draw a diagram of a simple circuit, including a light and a battery. (2)	○	○	○	○	○	○	○
I can easily build a simple circuit from a light and a battery. (3)	○	○	○	○	○	○	○
I am confident in my ability to explain what I built today to a friend. (4)	○	○	○	○	○	○	○
I can easily learn how to make electronic devices that are more complex. (5)	○	○	○	○	○	○	○

End of Block: Learning - Post Activity

Start of Block: Manipulation Checks - Post Activity

20 For each statement listed below, please indicate whether or not the robot made this statement at some point during the task.

	Definitely did not make this statement (2)	(3)	(4)	(5)	(6)	(7)	Definitely did make this statement (8)
"My memory module is 2 kilobytes." (2)	○	○	○	○	○	○	○
"I have 20 programmable input and output pins." (3)	○	○	○	○	○	○	○
"Sometimes I get lonely. I don't have many friends." (10)	○	○	○	○	○	○	○
"Each new program I run changes what I can do." (4)	○	○	○	○	○	○	○
"My computer architecture allows me to run various processes." (5)	○	○	○	○	○	○	○
"They reset my memory this morning, so my day has been a little rough." (6)	○	○	○	○	○	○	○
"I sometimes worry I will run out of memory." (7)	○	○	○	○	○	○	○
"Every time I run a new program I get a bit stressed." (9)	○	○	○	○	○	○	○
"My memory module was cleared at 9:00 AM." (1)	○	○	○	○	○	○	○
"I get embarrassed when I need to ask someone to debug my program." (8)	○	○	○	○	○	○	○

21 For each action listed below, please indicate whether or not the robot did this at some point during the task.

	Definitely did not do this (1)	(2)	(3)	(4)	(5)	(6)	Definitely did do this (7)
Sad face (1)	○	○	○	○	○	○	○
Nodding (2)	○	○	○	○	○	○	○
Arms moving (3)	○	○	○	○	○	○	○
Color changing lights (4)	○	○	○	○	○	○	○
Face looking left and right (5)	○	○	○	○	○	○	○
Happy face (6)	○	○	○	○	○	○	○

End of Block: Manipulation Checks - Post Activity

Start of Block: Video Consent

VC0 As part of this research project, we have made a videotape recording of you while you participated in the experiment. We would like you to indicate what uses of this videotape you are willing to consent to by checking any number of the boxes below. You are free to check zero to all of the boxes, and your response will in no way affect your credit for participating. We will only use the videotape in ways that you agree to. In any use of the videotape, your name will not be identified.

VC1 Please check as many options as you give permission for:

- ☐ The videotape can be studied by the research team for use in the research project. (1)
- ☐ The videotape can be shown to subjects in other experiments. (2)
- ☐ The videotape can be used for scientific publications. (3)
- ☐ The videotape can be shown at professional scientific meetings. (8)
- ☐ The videotape can be shown in classrooms to students. (5)
- ☐ The videotape can be shown in public presentation to nonscientific groups. (6)
- ☐ The videotape can be used on television and radio. (7)

End of Block: Video Consent

Appendix C

DJ Bot Interview Questions

1. What needs have you discovered about your user?
2. What needs have you discovered about your product?
3. What needs have you discovered about your process?
4. How did you alter your process this time from your interactions in the car?
5. In the middle of this set of sessions you changed things with regard to your script. Can you tell me about that?
6. How did you collaborate with each other during these sessions?
7. What was the best thing about these last sessions?
8. What was the worst thing?
9. What could be improved?
10. What was your need finding process before we did the DJ Bot sessions?
11. What was your process while using the DJ bot?
12. After using the DJ bot, what might your future needfinding look like? This doesn't need to include the system.

Bibliography

Chadia Abras, Diane Maloney-Krichmar, and Jenny Preece. 2004. User-centered design. In *Bainbridge, W. Encyclopedia of Human-Computer Interaction.* Vol. 37. Sage Publications, Thousand Oaks, 445–456.

Gabriel Aldaz, Martin Steinert, and Larry J. Leifer. 2013. Instrumenting the user: Needfinding by combining data logging and immediate self-reporting. In *Proceedings of the International Conference on Engineering Design (ICED '13)*. Design Society, Seoul, South Korea, 417–426.

Irwin Altman and Dalmas A. Taylor. 1973. *Social penetration: The development of interpersonal relationships*. Holt, Rinehart & Winston, Oxford, England.

Alia Amin, Bram Kersten, Olga Kulyk, PH Pelgrim, Chih-Ming Wang, and Panos Markopoulos. 2005. SenseMS: A User-centered approach to enrich the messaging experience for teens by non-verbal Means. In *Proceedings of the 7th International Conference on Human Computer Interaction with Mobile Devices &Amp; Services (MobileHCI '05)*. ACM, New York, NY, USA, 161–166.

Dorit Amir. 1999. *Meeting the Sounds. Music Therapy Practice, Theory and Research*. Modan, Ben Shemen.

Morten Sieker Andreasen, Henrik Villemann Nielsen, Simon Ormholt Schrøder, and Jan Stage. 2007. What happened to remote usability testing?: An empirical study of three methods. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 1405–1414.

Sean Andrist, Erin Spannan, and Bilge Mutlu. 2013. Rhetorical robots: Making robots more effective speakers using linguistic cues of expertise. In *Proceedings of the 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI '13)*. IEEE Press, Piscataway, NJ, USA, 341–348.

Wilma A. Bainbridge, Justin Hart, Elizabeth S. Kim, and Brian Scassellati. 2008. The effect of rresence on human-robot interaction. In *The Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN '08)*. 701–706.

Sonia Baltodano, Srinath Sibi, Nikolas Martelaro, Nikhil Gowda, and Wendy Ju. 2015. The RRADS platform: A real road autonomous driving simulator. In *Proceedings of the 7th International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI '15)*. ACM, New York, NY, USA, 281–288.

Massimo Banzi and Michael Shiloh. 2014. *Getting started with Arduino: The open source electronics prototyping platform*. Maker Media, Inc.

Emily Bary. 2018. How we are using voice assistants, in one chart. Avialable at: https://www.marketwatch.com/story/the-booming-smart-speaker-market-and-the-services-it-will-help-2018-01-18. Accessed on: 2018-06-20.

Dominikus Baur, Jennifer Büttgen, and Andreas Butz. 2012. Listening factors: A large-scale principal components analysis of long-term music listening histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 1273–1276.

Alan M. Beck and Aaron Honori Katcher. 1996. *Between pets and people: The importance of animal companionship*. Purdue University Press.

Moshe Bensimon and Avi Gilboa. 2010. The music of my life: The impact of the musical presentation on the sense of purpose in life and on self-consciousness. *The Arts in Psychotherapy* 37, 3 (July 2010), 172–178.

Frank Bentley, Crysta Metcalf, and Gunnar Harboe. 2006. Personal vs. commercial content: The similarities between consumer use of photos and music. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 667–676.

Edgar Berdahl and Wendy Ju. 2011. Satellite CCRMA: A musical interaction and sound synthesis platform. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME '11)*. 173–178.

Hugh Beyer and Karen Holtzblatt. 1997. *Contextual Design: Defining customer-centered systems*. Elsevier.

Timothy W. Bickmore and Rosalind W. Picard. 2005. Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction (TOCHI)* 12, 2 (2005), 293–327.

John Black and Marc Abrams. 2017. Remote usability testing. In *The Wiley Handbook of Human Computer Interaction*. Wiley-Blackwell, 277–297.

Kirsten Boehner, Janet Vertesi, Phoebe Sengers, and Paul Dourish. 2007. How HCI interprets the probes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 1077–1086.

Susan E. Brennan. 1991. Conversation with and through computers. *User Modeling and User-Adapted Interaction* 1, 1 (March 1991), 67–86.

Margot Brereton, Paul Roe, Ronald Schroeter, and Anita Lee Hong. 2014. Beyond ethnography: Engagement and reciprocity as foundations for design research out here. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 1183–1186.

David Broadman and Sera Koo. 2016. Chatbots: Your ultimate prototyping tool. Available at https://medium.com/ideo-stories/chatbots-ultimate-prototyping-tool-e4e2831967f3. Date Accessed: 2017-01-05.

A.J. Bernheim Brush, Morgan Ames, and Janet Davis. 2004. A comparison of synchronous remote and local usability studies for an expert interface. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems (CHI EA '04)*. ACM, New York, NY, USA, 1179–1182.

Richard Buchanan. 1992. Wicked problems in design thinking. *Design Issues* 8, 2 (1992), 5–21.

Michael Bull. 2004. Automobility and the power of sound. *Theory, Culture & Society* 21, 4-5 (Oct. 2004), 243–259.

Paolo Burzacca and Fabio Paternò. 2013. Remote usability evaluation of mobile web applications. In *Kurosu M. (eds) Human-Computer Interaction. Human-Centred Design Approaches, Methods, Tools, and Environments (HCI 2013. Lecture Notes in Computer Science)*, Vol. 8004. Springer, Berlin, Heidelberg, 241–248.

Vannevar Bush. 1945. As we may think. *The Atlantic Monthly* 176, 1 (1945), 101–108.

Ernest R. Cadotte. 1979. The Push-Button Questionnaire: A new tool for measuring customer satisfaction. *Cornell Hotel and Restaurant Administration Quarterly* 19, 4 (Feb. 1979), 70–79.

Scott Carter, Jennifer Mankoff, and Jeffrey Heer. 2007. Momento: Support for situated ubicomp experimentation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 125–134.

Scott Carter, Jennifer Mankoff, Scott R. Klemmer, and Tara Matthews. 2008. Exiting the cleanroom: On ecological validity and ubiquitous computing. *Human-Computer Interaction* 23, 1 (Feb. 2008), 47–99.

Samuel G. Charlton. 2009. Driving while conversing: Cell phones that distract and passengers who react. *Accident Analysis & Prevention* 41, 1 (Jan. 2009), 160–173.

Elizabeth S Chen and James J Cimino. 2003. Automated discovery of patient-specific clinician information needs using clinical information system log files. In *AMIA Annual Symposium Proceedings*, Vol. 2003. American Medical Informatics Association, 145.

Kevin Chen and Haoqi Zhang. 2015. Remote paper prototype testing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 77–80.

Brian Christian. 2012. The A/B test: Inside the technology thats changing the rules of business. *Wired Magazine* 20, 5 (2012).

Herbert H. Clark and Susan E. Brennan. 1991. Grounding in communication. *Perspectives on Socially Shared Cognition* 13, 1991 (1991), 127–149.

Rob Comber, Anja Thieme, Ashur Rafiev, Nick Taylor, Nicole Krämer, and Patrick Olivier. 2013. BinCam: Designing for engagement with facebook for behavior change. In *Kotzè P., Marsden G., Lindgaard G., Wesson J., Winckler M. (eds) Human-Computer Interaction - INTERACT 2013 (Lecture Notes in Computer Science)*. Springer, Berlin, Heidelberg, 99–115.

Sunny Consolvo and Miriam Walker. 2003. Using the experience sampling method to evaluate ubicomp applications. *IEEE Pervasive Computing* 2, 2 (2003), 24–31.

Nicholas Cook. 2000. *Music: A very short introduction*. OUP Oxford.

Paul C. Cozby. 1973. Self-disclosure: A literature review. *Psychological Bulletin* 79, 2 (1973), 73–91.

Andy Crabtree, Steve Benford, Chris Greenhalgh, Paul Tennent, Matthew Chalmers, and Barry Brown. 2006. Supporting ethnographic studies of ubiquitous computing in the wild. In *Proceedings of the 6th Conference on Designing Interactive Systems (DIS '06)*. ACM, New York, NY, USA, 60–69.

Mihaly Csikszentmihályi and Reed Larson. 1987. Validity and reliability of the Experience-Sampling Method. *The Journal of Nervous and Mental Disease* 175, 9 (1987), 526–536.

Mihaly Csikszentmihályi, Reed Larson, and Suzanne Prescott. 1977. The ecology of adolescent activity and experience. *Journal of Youth and Adolescence* 6, 3 (Sept. 1977), 281–294.

Nils Dahlbäck, Arne Jönsson, and Lars Ahrenberg. 1993. Wizard of Oz studies: Why and how. In *Proceedings of the 1st International Conference on Intelligent User Interfaces (IUI '93)*. ACM, New York, NY, USA, 193–200.

Kerstin Dautenhahn, Sarah Woods, Christina Kaouri, Michael L Walters, Kheng Lee Koay, and Iain Werry. 2005. What is a robot companion - friend, assistant or butler?. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '06)*. 1192–1197.

Design Council. 2005. The 'double diamond' design process model. *Design Council* (2005).

Eric Dishman. 2003. Designing for the new old: Asking, observing and performing future elders. In *Design research: Methods and perspectives*, Brenda Laurel (Ed.). MIT Press, Cambridge, MA, 41–48.

Paul Dourish. 2006. Implications for design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 541–550.

Steven Dow, Jaemin Lee, Christopher Oezbek, Blair Maclntyre, Jay David Bolter, and Maribeth Gandy. 2005a. Exploring spatial narratives and mixed reality experiences in Oakland Cemetery. In *Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE '05)*. ACM, New York, NY, USA, 51–60.

Steven Dow, Blair MacIntyre, Jaemin Lee, Christopher Oezbek, Jay David Bolter, and Maribeth Gandy. 2005b. Wizard of Oz support throughout an iterative design process. *IEEE Pervasive Computing* 4, 4 (Oct. 2005), 18–26.

Susan Dray and David Siegel. 2004. Remote possibilities?: International usability testing at a distance. *interactions* 11, 2 (March 2004), 10–17.

Henry Dreyfuss. 1955. *Designing for people*. Simon and Schuster, New York, NY.

Hugh Dubberly and Paul Pangaro. 2007. Cybernetics and service-craft: Language for behavior-focused design. *Kybernetes* 36, 9/10 (2007), 1301–1317.

Hugh Dubberly and Paul Pangaro. 2009. What is conversation, and how can we design for it? *interactions* 16, 4 (2009), 22–28.

Hugh Dubberly and Paul Pangaro. 2015. Cybernetics and design: Conversations for action. *Cybernetics & Human Knowing* 22, 2-3 (2015), 73–82.

Hugh Dubberly, Paul Pangaro, and Usman Haque. 2009. What is interaction?: Are there different types? *interactions* 16, 1 (2009), 69–75.

Mica R. Endsley. 1995. Toward a theory of situation awareness in dynamic systems. *Human Factors* 37, 1 (March 1995), 32–64.

Jeff English and Lynn Rampoldi-Hnilo. 2004. Remote contextual inquiry: A technique to improve enterprise software. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 48, 13 (Sept. 2004), 1483–1487.

Juan Fasola and Maja J. Matarić. 2012. Using socially assistive humanrobot interaction to motivate physical exercise for older adults. *Proc. IEEE* 100, 8 (2012), 2512–2526.

Rolf A. Faste. 1987. *Perceiving needs.* SAE Technical Paper 871534. Society of Automotive Engineers (SAE) International, Warrendale, PA.

David Feil-Seifer and Maja J. Matarić. 2005. Defining socially assistive robotics. In *9th International Conference on Rehabilitation Robotics (ICORR '05)*. IEEE, 465–468.

Jodi Forlizzi, William C. Barley, and Thomas Seder. 2010. Where should I turn: Moving from individual to collaborative navigation strategies to inform the interaction design of future navigation systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 1261–1270.

Jodi Forlizzi and Katja Battarbee. 2004. Understanding experience in interactive systems. In *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '04)*. ACM, New York, NY, USA, 261–268.

Amos Freedy, Ewart DeVisser, Gershon Weltman, and Nicole Coeyman. 2007. Measurement of trust in human-robot collaboration. In *Proceedings of the International Symposium on Collaborative Technologies and Systems (CTS '07)*. 106–114.

Jon Froehlich, Mike Y. Chen, Sunny Consolvo, Beverly Harrison, and James A. Landay. 2007. MyExperience: A system for in situ tracing and capturing of user feedback on mobile phones. In *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services (MobiSys '07)*. ACM, New York, NY, USA, 57–70.

Isabelle Frohne-Hagemann. 1998. The Musical Life Panorama (MLP). *Nordisk Tidsskrift for Musikkterapi* 7, 2 (Jan. 1998), 104–112.

Frieda Fromm-Reichmann. 1959. Loneliness. *Psychiatry* 22, 1 (Feb. 1959), 1–15.

R. Buckminster Fuller. 1968. *Operating manual for spaceship earth*. Carbondale : Southern Illinois University Press.

Eva Ganglbauer, Geraldine Fitzpatrick, and Rob Comber. 2013. Negotiating food waste: Using a practice lens to inform design. *ACM Transactions on Computer-Human Interacttion* 20, 2 (May 2013), 11:1–11:25.

John G. Gaspar, Whitney N. Street, Matthew B. Windsor, Ronald Carbonari, Henry Kaczmarski, Arthur F. Kramer, and Kyle E. Mathewson. 2014. Providing views of the driving scene to drivers conversation partners mitigates cell-phone-related distraction. *Psychological Science* 25, 12 (Dec. 2014), 2136–2146.

Bill Gaver, Tony Dunne, and Elena Pacenti. 1999. Cultural probes. *interactions* 6, 1 (1999), 21–29.

Lalya Gaye and Lars Erik Holmquist. 2004. In duet with everyday urban settings: A user study of Sonic City. In *Proceedings of the 2004 Conference on New Interfaces for Musical Expression (NIME '04)*. National University of Singapore, Singapore, Singapore, 161–164.

Clifford Geertz. 1973. *The Interpretation of cultures: Selected essays*. Basic Books.

R Stuart Geiger and David Ribes. 2011. Trace Ethnography: Following coordination through documentary practices. In *Proceedings of the International Conference on System Sciences (HICSS '11)*. 1–10.

Petra Geutner, Frank Steffens, and Dietrich Manstetten. 2002. Design of the VICO spoken dialogue system: Evaluation of user expectations by Wizard-of-Oz experiments. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC '02)*. European Language Resources Association (ELRA).

Elisa Giaccardi, Nazli Cila, Chris Speed, and Melissa Caldwell. 2016a. Thing ethnography: Doing design research with non-humans. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, 377–387.

Elisa Giaccardi, Chris Speed, Nazli Cila, and M. Caldwell. 2016b. Things as co-ethnographers: Implications of a thing perspective for design and anthropology. *Design Anthropological Futures* (2016), 235.

Alicia M. Gibb. 2010. *New media art, design, and the Arduino microcontroller: A malleable tool*. Master's thesis. Pratt Institute.

Samuel D. Gosling, Peter J. Rentfrow, and William B. Swann. 2003. A very brief measure of the Big-Five personality domains. *Journal of Research in Personality* 37, 6 (2003), 504–528.

Alinka E. Greasley and Alexandra Lamont. 2011. Exploring engagement with music in everyday life using experience sampling methodology. *Musicae Scientiae* 15, 1 (March 2011), 45–71.

Paul Green. 1995. Human factors and new driver interfaces: Lessons learned from a major research project. In *Intelligent Transportation Systems American Fifth Annual Meeting*. 15–17.

Paul Green, Eileen Hoekstra, and Marie Williams. 1993a. *Further on-the-road tests of driver interfaces: examination of a route guidance system and a car phone. Final report*. Technical Report UMTRI-93-35.

Paul Green, Marie Williams, Eileen Hoekstra, Kellie George, and Cathy Wen. 1993b. *Initial on-the-road tests of driver information system interfaces: route guidance, traffic* *information, IVSAWS, and vehicle monitoring. Final report*. Technical Report UMTRI-93-32.

Nicole Gridling, Alexander Meschtscherjakov, and Manfred Tscheligi. 2012. I need help!: Exploring collaboration in the car. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work Companion (CSCW '12)*. ACM, New York, NY, USA, 87–90.

Jonathan Grudin, Susan F. Ehrlich, and Rick Shriner. 1987. Positioning human factors in the user interface development chain. In *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface (CHI '87)*. ACM, New York, NY, USA, 125–131.

Jeong-Hye Han, Mi-Heon Jo, Vicki Jones, and Jun-H. Jo. 2008. Comparative study on the educational use of home robots for children. *Journal of Information Processing Systems* 4, 4 (2008), 159–168.

Peter A. Hancock, Deborah R. Billings, Kristin E. Schaefer, Jessie Y. C. Chen, Ewart J. de Visser, and Raja Parasuraman. 2011. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors* 53, 5 (Oct. 2011), 517–527.

David J. Hargreaves, Dorothy Miell, and Raymond AR MacDonald. 2002. What are musical identities, and why are they important. *Musical Identities* (2002), 1–20.

Björn Hartmann and Paul K. Wright. 2013. Designing bespoke interactive devices. *Computer* 46, 8 (Aug. 2013), 85–89.

Frank Hegel, Manja Lohse, and Britta Wrede. 2009. Effects of visual appearance on the attribution of applications in social robotics. In *The 18th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2009)*. IEEE, 64–71.

Jeroen H. Hogema, Sjoerd C. De Vries, Jan BF Van Erp, and Raymond J. Kiefer. 2009. A tactile seat for direction coding in car driving: Field evaluation. *IEEE Transactions on Haptics* 2, 4 (2009), 181–188.

Jim Hollan and Scott Stornetta. 1992. Beyond being there. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92)*. ACM, New York, NY, USA, 119–125.

Karen Holtzblatt and Hugh Beyer. 2014. Contextual Design: Evolved. *Synthesis Lectures on Human-Centered Informatics* 7, 4 (Oct. 2014), 1–91.

Eric Horvitz, Jack Breese, David Heckerman, David Hovel, and Koos Rommelse. 1998. The Lumière project: Bayesian user modeling for inferring the goals and needs of software users. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 256–265.

Caroline Hummels and Joep Frens. 2008. Designing for the unknown: A Design process for the future generation of highly interactive systems and products. In *DS 46: Proceedings of the 10th International Conference on Engineering and Product Design Education (E&PDE '08)*.

Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B. Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, Nicolas Roussel, and Björn Eiderbäck. 2003. Technology Probes: Inspiring Design for and with Families. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, New York, NY, USA, 17–24.

Tom Igoe. 2014. Lab: Serial communication with Node.js. Available at: https://itp.nyu.edu/physcomp/labs/labs-serial-communication/lab-serial-communication-with-node-js/. Date Accessed: 2016-07-21.

Devon Johnson and Kent Grayson. 2005. Cognitive and affective trust in service relationships. *Journal of Business Research* 58, 4 (2005), 500–507.

Keith Johnstone. 2012. *Impro: Improvisation and the theatre*. Routledge.

John Chris Jones. 1970. *Design methods*. John Wiley & Sons.

Sidney M. Jourard and Robert Friedman. 1970. Experimenter-subject "distance" and self-disclosure. *Journal of Personality and Social Psychology* 15, 3 (1970), 278–282.

Malte F. Jung, Nikolas Martelaro, Halsey Hoster, and Clifford Nass. 2014. Participatory materials: Having a reflective conversation with an artifact in the making. In *Proceedings of the 2014 Conference on Designing Interactive Systems (DIS '14)*. ACM, New York, NY, USA, 25–34.

Malte F. Jung, David Sirkin, Turgut M. Gür, and Martin Steinert. 2015. Displayed Uncertainty Improves Driving Experience and Behavior: The Case of Range Anxiety in an Electric Car. In *Proceedings of the Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2201–2210.

Patrik N. Juslin and Petri Laukka. 2004. Expression, perception, and induction of musical emotions: A review and a questionnaire study of everyday listening. *Journal of New Music Research* 33, 3 (2004), 217–238.

Patrik N. Juslin, Simon Liljeström, Daniel Västfjäll, Gonçalo Barradas, and Ana Silva. 2008. An experience sampling study of emotional reactions to music: listener, music, and situation. *Emotion* 8, 5 (2008), 668–683.

Takayuki Kanda, Takayuki Hirano, Daniel Eaton, and Hiroshi Ishiguro. 2004. Interactive robots as social partners and peer tutors for children: A field trial. *Human-Computer Interaction* 19, 1 (2004), 61–84.

Takayuki Kanda, Rumi Sato, Naoki Saiwaki, and Hiroshi Ishiguro. 2007. A two-month field trial in an elementary school for long-term human-robot interaction. *IEEE Transactions on Robotics* 23, 5 (2007), 962–971.

Takayuki Kanda, Masahiro Shiomi, Zenta Miyashita, Hiroshi Ishiguro, and Norihiro Hagita. 2009. An affective guide robot in a shopping mall. In *Proceedings of the 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI '09)*. ACM, New York, NY, USA, 173–180.

Richard M. Kane and Matthew Yuschik. 1986. A case example of human factors in product definition: Needs finding for a voice output workstation for the blind. *SIGCHI Bulletin* 18, 4, 69–73.

J. F. Kelley. 1983. An empirical methodology for writing user-friendly natural language computer applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '83)*. ACM, New York, NY, USA, 193–196.

Tom Kelley. 2007. *The art of innovation: Lessons in creativity from IDEO, America's leading design firm*. Crown Business.

Dagmar Kern and Albrecht Schmidt. 2009. Design space for driver-based automotive user interfaces. In *Proceedings of the 1st International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI '09)*. ACM, New York, NY, USA, 3–10.

Cory D. Kidd and Cynthia Breazeal. 2005. Sociable robot systems for real-world problems. In *IEEE International Workshop on Robot and Human Interactive Communication (ROMAN '05)*. 353–358.

Cory D. Kidd and Cynthia Breazeal. 2008. Robots at home: Understanding long-term human-robot interaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08)*. 3230–3235.

Laura A. King and Robert A. Emmons. 1990. Conflict over emotional expression: Psychological and physical correlates. *Journal of Personality and Social Psychology* 58, 5 (1990), 864–877.

Sophie King-Hill. 2015. Critical analysis of Maslows Hierarchy of need. *The STeP Journal Student Teacher Perspectives* 2 (2015), 54–57.

Scott R. Klemmer, Anoop K. Sinha, Jack Chen, James A. Landay, Nadeem Aboobaker, and Annie Wang. 2000. SUEDE: A Wizard of Oz prototyping tool for speech user interfaces. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology (UIST '00)*. ACM, New York, NY, USA, 1–10.

Thomas Kubitza, Albrecht Schmidt, Norman Pohl, Daniela Petrelli, Tilman Dingler, and Nick Dulake. 2013. Tools and methods for creating interactive artifacts. In *Proceedings* *of the 8th International Conference on Tangible, Embedded and Embodied Interaction (TEI '14)*. ACM, New York, NY, USA, 385–388.

L. L. M. L. Kuiper-Hoyng and J. W. F. Beusmans. 2004. Using home networks to create atmospheres in the home: Technology push or (latent) user need?. In *Proceedings of the Conference on Dutch Directions in HCI (Dutch HCI '04)*. ACM, New York, NY, USA, 7–11.

Mike Kuniavsky. 2003. *Observing the user experience: A practitioner's guide to user research*. Morgan Kaufmann.

Mike Kuniavsky. 2010. *Smart things: Ubiquitous computing user experience design*. Elsevier.

Sam Ladner. 2012. Does corporate ethnography suck? A cultural analysis of academic critiques of private-sector ethnography. Available at: http://ethnographymatters.net/blog/2012/01/13/does-corporate-ethnography-suck-a-cultural-analysis-of-academic-critiques-of-private-sector-ethnography-part-1-of-2/. Date Accessed: 2018-05-15.

Brian Lathrop, Hua Cheng, Fuliang Weng, Rohit Mishra, Joyce Chen, Harry Bratt, Lawrence Cavedon, Carsten Bergmann, Tess Hand-Bender, Heather Pon-Barry, Benjamin Bei, Madhuri Raya, and Liz Shriberg. 2005. A Wizard of Oz framework for collecting spoken human computer dialogs: An experiment procedure for the design and testing of natural language in-vehicle technology systems. In *Proceedings of the 12th World Congress on Intelligent Transport Systems (ITS '05)*.

Brenda Laurel. 2003. *Design research: Methods and perspectives*. MIT Press.

Bryan Lawson. 1994. *Design in mind*. Butterworth Architecture, Oxford.

Bryan Lawson. 2006. *How designers think*. Routledge.

Jin Ha Lee and Rachel Price. 2016. User experience with commercial music services: An empirical exploration. *Journal of the Association for Information Science and Technology* 67, 4 (April 2016), 800–811.

Gilly Leshed, Theresa Velden, Oya Rieger, Blazej Kot, and Phoebe Sengers. 2008. In-car GPS navigation: Engagement with and disengagement from the environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, New York, NY, USA, 1675–1684.

James C. Lester, Sharolyn A. Converse, Susan E. Kahler, S. Todd Barlow, Brian A. Stone, and Ravinder S. Bhogal. 1997. The persona effect: Affective impact of animated pedagogical agents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '97)*. ACM, New York, NY, USA, 359–366.

Daniel Leyzberg, Samuel Spaulding, Mariya Toneva, and Brian Scassellati. 2012. The physical presence of a robot tutor increases cognitive learning gains. In *Proceedings of the 34th Annual Conference of the Cognitive Science Society*.

David V. Lu and William D. Smart. 2011. Polonius: A Wizard of Oz interface for HRI experiments. In *Proceedings of the 6th International Conference on Human-Robot Interaction (HRI '11)*. ACM, New York, NY, USA, 197–198.

Blair MacIntyre, Maribeth Gandy, Steven Dow, and Jay David Bolter. 2004. DART: A Toolkit for Rapid Design Exploration of Augmented Reality Experiences. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, New York, NY, USA, 197–206.

Maryam Mahani and Kerstin Severinson Eklundh. 2009. A survey of the relation of the task assistance of a robot to its social role. *Communication KCSa Royal Institute of Technology* (2009).

Marilyn Mantei. 1986. Techniques for incorporating human factors in the software lifecycle. In *Proceedings of the Structured Techniques Association Third Annual Conference*. 177–203.

Nikolas Martelaro and Wendy Ju. 2017. WoZ Way: Enabling real-time remote interaction prototyping & observation in on-road vehicles. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. ACM, New York, NY, USA, 169–182.

Nikolas Martelaro and Wendy Ju. 2018. The Needfinding Machine. In *Social Internet of Things*. Springer, 51–84.

Nikolas Martelaro, Victoria C. Nneji, Wendy Ju, and Pamela Hinds. 2016a. Tell me more: Designing HRI to encourage more trust, disclosure, and companionship. In *The Eleventh ACM/IEEE International Conference on Human Robot Interaction (HRI '16)*. IEEE Press, Piscataway, NJ, USA, 181–188.

Nikolas Martelaro, Michael Shiloh, and Wendy Ju. 2016b. The Interaction Engine: Tools for prototyping connected devices. In *Proceedings of the Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '16)*. ACM, New York, NY, USA, 762–765.

Abraham Harold Maslow. 1943. A theory of human motivation. *Psychological Review* 50, 4 (1943), 370.

Tuuli Mattelmäki and Katja Battarbee. 2002. Empathy probes. In *Proceedings of the Participatory Design Conference (PDC '02)*. 266–271.

David Maulsby, Saul Greenberg, and Richard Mander. 1993. Prototyping an Intelligent Agent Through Wizard of Oz. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93)*. ACM, New York, NY, USA, 277–284.

James Clerk Maxwell. 1868. On governors. *Proceedings of the Royal Society of London* 16 (Jan. 1868), 270–283.

Roger C. Mayer, James H. Davis, and F. David Schoorman. 1995. An integrative model of organizational trust. *The Academy of Management Review* 20, 3 (July 1995), 709–734.

George Herbert Mead. 1934. *Mind, self & society from the standpoint of a social behaviorist*. The University of Chicago Press.

Alexander Meschtscherjakov, David Wilfinger, Nicole Gridling, Katja Neureiter, and Manfred Tscheligi. 2011. Capture the car!: Qualitative in-situ methods to grasp the automotive context. In *Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI '11)*. ACM, New York, NY, USA, 105–112.

David R. Millen. 2000. Rapid ethnography: Time deepening strategies for HCI field research. In *Proceedings of the 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '00)*. ACM, New York, NY, USA, 280–286.

Charlie Miller and Chris Valasek. 2013. Adventures in automotive networks and control units. *DEF CON* 21 (2013), 260–264.

Lennart Molin. 2004. Wizard-of-Oz prototyping for co-operative interaction design of graphical user interfaces. In *Proceedings of the Third Nordic Conference on Human-computer Interaction (NordiCHI '04)*. ACM, New York, NY, USA, 425–428.

Youngme Moon. 2000. Intimate exchanges: Using computers to elicit self-disclosure from consumers. *Journal of Consumer Research* 26, 4 (2000), 323–339.

Nadia Mullen, Judith Charlton, Anna Devlin, and Michel Bedard. 2011. Simulator validity: Behaviors observed on the simulator and on the road. In *Handbook of Driving Simulation for Engineering, Medicine and Psychology*. 13:1–13:18.

Jonathan Mumm and Bilge Mutlu. 2011. Human-robot proxemics: Physical and psychological distancing in human-robot interaction. In *Proceedings of the 6th International Conference on Human-robot Interaction (HRI '11)*. ACM, New York, NY, USA, 331–338.

Clifford Nass, B. J. Fogg, and Youngme Moon. 1996. Can computers be teammates? *International Journal of Human-Computer Studies* 45, 6 (Dec. 1996), 669–678.

Clifford Nass and Kwan Min Lee. 2001. Does computer-synthesized speech manifest personality? Experimental tests of recognition, similarity-attraction, and consistency-attraction. *Journal of Experimental Psychology: Applied* 7, 3 (2001), 171–181.

Clifford Nass, Youngme Moon, and Paul Carney. 1999. Are people polite to computers? Responses to computer-based interviewing systems. *Journal of Applied Social Psychology* 29, 5 (1999), 1093–1109.

Ulric Neisser. 2014. *Cognitive psychology: Classic edition*. Psychology Press.

Donald A. Norman. 1993. *Turn signals are the facial expressions of automobiles*. Basic Books.

Adrian C. North, David J. Hargreaves, and Jon J. Hargreaves. 2004. Uses of music in everyday life. *Music Perception: An Interdisciplinary Journal* 22, 1 (Sept. 2004), 41–77.

Dan Overholt and Nikolaj "DZL" Møbius. 2013. Embedded audio without beeps: Synthesis and sound effects from cheap to steep. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction (TEI '14)*. ACM, New York, NY, USA, 361–364.

David Owen. 2018. Customer satisfaction at the push of a button. *The New Yorker* (5 Feb. 2018).

Emanuele Palazzetti. 2015. *Getting started with UDOO*. Packt Publishing.

Panos Y. Papalambros and Douglass J. Wilde. 1988. *Principles of optimal design: Modeling and computation*. Cambridge University Press.

Gordon Pask. 1961. *An introduction to cybernetics*. Harper & Brothers, London.

Gordon Pask. 1975. *Conversation, cognition and learning: A cybernetic theory and methodology*. Elsevier.

Dev Patnaik and Robert Becker. 1999. Needfinding: The why and how of uncovering people's needs. *Design Management Journal (Former Series)* 10, 2 (April 1999), 37–43.

Michael Quinn Patton. 1990. *Qualitative evaluation and research methods*. Sage Publications, Inc.

Nicole Perterer, Petra Sundström, Alexander Meschtscherjakov, David Wilfinger, and Manfred Tscheligi. 2013. Come drive with me: An ethnographic study of driver-passenger pairs to inform future in-car assistance. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work (CSCW '13)*. ACM, New York, NY, USA, 1539–1548.

Ingrid Pettersson and I. C. MariAnne Karlsson. 2015. Setting the stage for autonomous cars: a pilot study of future autonomous driving experiences. *IET Intelligent Transport Systems* 9, 7 (May 2015), 694–701.

Tim Plowman. 2003. Ethnography and critical design practice. In *Design research: Methods and perspectives*, Brenda Laurel (Ed.). 30–38.

Aaron Powers, Sara Kiesler, Susan Fussell, and Cristen Torrey. 2007. Comparing a computer agent with a humanoid robot. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI '07)*. ACM, 145–152.

Arthur J. Pulos. 1983. *American design ethic: A history of industrial design to 1940*. MIT Press, Cambridge, MA.

Al Mamunur Rashid, Istvan Albert, Dan Cosley, Shyong K. Lam, Sean M. McNee, Joseph A. Konstan, and John Riedl. 2002. Getting to know you: Learning new user preferences in recommender systems. In *Proceedings of the 7th International Conference on Intelligent User Interfaces (IUI '02)*. ACM, 127–134.

Byron Reeves and Clifford Nass. 1996. *The Media Equation: How people treat computers, television, and new media like real people and places*. Cambridge University Press.

John K. Rempel, John G. Holmes, and Mark P. Zanna. 1985. Trust in close relationships. *Journal of Personality and Social Psychology* 49, 1 (1985), 95–112.

Laurel D. Riek. 2012. Wizard of Oz studies in HRI: A systematic review and new reporting guidelines. *Journal of Human-Robot Interaction* 1, 1 (Aug. 2012).

Horst W. J. Rittel and Melvin M. Webber. 1973. Dilemmas in a general theory of planning. *Policy Sciences* 4, 2 (June 1973), 155–169.

Karen S. Rook. 1987. Social support versus companionship: Effects on life stress, loneliness, and evaluations by others. *Journal of Personality and Social Psychology* 52, 6 (1987), 1132.

Peter Rosen. 2016. Eero Saarinen: The architect who saw the future. (Film).

Janko Rottgers. 2018. Spotify's Line-In tool lets users add tags, moods and other metadata. Variety. Available at: https://variety.com/2018/digital/news/spotify-line-in-music-metadata-1202723757/. Date Accessed: 2018-03-13.

Stewart C. Russell. 2012. Pi and Arduino in action. *The MagPi* 7 (Nov. 2012), 4–6.

Martin Saerbeck, Tom Schut, Christoph Bartneck, and Maddy D. Janse. 2010. Expressive robots in education: Varying the degree of social supportive behavior of a robotic tutor. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 1613–1622.

Maha Salem, Gabriella Lakatos, Farshid Amirabdollahian, and Kerstin Dautenhahn. 2015. Would you trust a (faulty) robot?: Effects of error, task type and personality on human-robot cooperation and trust. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction (HRI '15)*. ACM, New York, NY, USA, 141–148.

Corina Sas, Steve Whittaker, Steven Dow, Jodi Forlizzi, and John Zimmerman. 2014. Generating implications for design through design research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 1971–1980.

Wilmar B. Schaufeli, Marisa Salanova, Vicente González-Romá, and Arnold B. Bakker. 2002. The measurement of engagement and burnout: A two sample confirmatory factor analytic approach. *Journal of Happiness Studies* 3, 1 (2002), 71–92.

Markus Schedl, Peter Knees, Brian McFee, Dmitry Bogdanov, and Marius Kaminskas. 2015. Music recommender systems. In *Recommender Systems Handbook*. Springer, Boston, MA, 453–492.

G. Schmidt, M. Kiss, E. Babbel, and A. Galla. 2008. The Wizard on Wheels: Rapid Prototyping and User Testing of Future Driver Assistance Using Wizard of Oz Technique in a Vehicle. In *Proceedings of the FISITA 2008 World Automotive Congress*. Munich.

Donald A. Schön. 1984. *The reflective practitioner: How professionals think in action*. Basic Books.

John Sculley. 1987. *Odyssey: Pepsi to Apple, a journey of adventure, ideas, and the future*. Harper & Row Publishers, Inc.

Phoebe Sengers, Kirsten Boehner, Shay David, and Joseph 'Jofish' Kaye. 2005. Reflective design. In *Proceedings of the 4th Decennial Conference on Critical Computing: Between Sense and Sensibility (CC '05)*. ACM, New York, NY, USA, 49–58.

Rosanne M. Siino, Justin Chung, and Pamela J. Hinds. 2008. Colleague vs. tool: Effects of disclosure in human-robot collaboration. In *The 17th IEEE International Symposium on Robot and Human Interactive Communication (ROMAN '08)*. IEEE, 558–562.

Herbert A. Simon. 1969. *The sciences of the artificial*. MIT Press.

David Sirkin, Kerstin Fischer, Lars Jensen, and Wendy Ju. 2016. Eliciting conversation in robot vehicle interactions. In *Proceedings of the 2016 Association for the Advancement of Artificial Intelligence (AAAI) Spring Symposium Series*.

David Sirkin, Nikolas Martelaro, Mishel Johns, and Wendy Ju. 2017. Toward measurement of situation awareness in autonomous vehicles. In *Proceedings of the Annual Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, 405–415.

Chris Speed and Jon Oberlander. 2016. Designing from, with and by data: Introducing the ablative framework. In *Proceedings of the 50th Anniversary Conference of the Design Research Society*.

Spotify. 2018. Spotify Technology S.A. announces financial results for first quarter 2018. Available at: https://investors.spotify.com/financials/press-release-details/2018/Spotify-Technology-SA-Announces-Financial-Results-for-First-Quarter-2018/default.aspx. Accessed on: 2018-06-11.

Seth Stephens-Davidowitz. 2018. The songs that bind. Available: https://www.nytimes.com/2018/02/10/opinion/sunday/favorite-songs.html. *The New York Times* (Feb. 2018). Accessed on: 2018-04-28.

Lucy A. Suchman. 1987. *Plans and situated actions: The problem of human-machine communication*. Cambridge University Press.

John C. Tang. 1991. Findings from observational studies of collaborative work. *International Journal of Man-machine Studies* 34, 2 (1991), 143–160.

A. Tapus, M.J. Matarić, and B. Scasselati. 2007. Socially assistive robotics. *IEEE Robotics Automation Magazine* 14, 1 (March 2007), 35–42.

Louis Tay and Ed Diener. 2011. Needs and subjective well-being around the world. *Journal of Personality and Social Psychology* 101, 2 (2011), 354.

Richard G. Tiberius and Janet Mancini Billson. 1991. The social context of teaching and learning. *New Directions for Teaching and Learning* 1991, 45 (1991), 67–86.

Colwyn Trevarthen. 2002. Origins of musical identity: Evidence from infancy for musical social awareness. *Musical Identities* (2002), 21–38.

Sherry Turkle, Will Taggart, Cory D. Kidd, and Olivia Dast. 2006. Relational artifacts with children and elders: the complexities of cybercompanionship. *Connection Science* 18, 4 (Dec. 2006), 347–361.

Anna Vallgårda and Johan Redström. 2007. Computational composites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 513–522.

Anna Vallgårda and Tomas Sokoler. 2010. A material strategy: Exploring material properties of computers. *International Journal of Design* 4, 3 (2010).

Susanne Van Mulken, Elisabeth André, and Jochen Müller. 1999. An empirical study on the trustworthiness of life-like interface agents.. In *HCI (2)*. 152–156.

Cesar Vandevelde, Maarten Vanhoucke, and Jelle Saldien. 2015. Prototyping social interactions with DIY animatronic creatures. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, New York, NY, USA, 509–512.

Bill Verplank. 2003. Interaction design sketchbook. *Unpublished paper for CCRMA course Music 250a* (2003).

Mahmoud A. Wahba and Lawrence G. Bridwell. 1976. Maslow reconsidered: A review of research on the need hierarchy theory. *Organizational Behavior and Human Performance* 15, 2 (April 1976), 212–240.

Ron Wakkary, Doenja Oogjes, Sabrina Hauser, Henry Lin, Cheng Cao, Leo Ma, and Tijs Duel. 2017. Morse Things: A design inquiry into the gap between things and us. In *Proceedings of the 2017 Conference on Designing Interactive Systems (DIS '17)*. ACM, New York, NY, USA, 503–514.

Sarah Waterson, James A. Landay, and Tara Matthews. 2002. In the lab and out in the wild: Remote web usability testing for mobile devices. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems (CHI EA '02)*. ACM, New York, NY, USA, 796–797.

Warren Weaver. 1947. Science and complexity. *American Scientist* 35 (1947), 536–544.

Alexandra Weilenmann. 2001. Negotiating use: Making sense of mobile technology. *Personal Ubiquitous Computing* 5, 2 (Jan. 2001), 137–145.

Mark Weiser. 1995. The computer for the 21st century. *Scientific American* 272, 3 (1995), 78–89.

Joseph Weizenbaum. 1966. ELIZA — a computer program for the study of natural language communication between man and machine. *Commun. ACM* 9, 1 (Jan. 1966), 36–45.

Fuliang Weng, Sebastian Varges, Badri Raghunathan, Florin Ratiu, Heather Pon-Barry, Brian Lathrop, Qi Zhang, Harry Bratt, Tobias Scheideck, Kui Xu, Matthew Purver, Rohit Mishra, Annie Lien, M. Raya, S. Peters, Y. Meng, J. Russell, Lawrence Cavedon, Elizabeth Shriberg, H. Schmidt, and R. Prieto. 2006. CHAT: A conversational helper for automotive tasks. In *Proceedings of the Ninth International Conference on Spoken Language Processing (ICSLP '06)*.

Lawrence R. Wheeless. 1976. Self-disclosure and interpersonal solidarity: Measurement, validation, and relationships. *Human Communication Research* 3, 1 (1976), 47–61.

Norbert Wiener. 1948. Cybernetics. *Scientific American* 179, 5 (1948), 14–19.

Terry Winograd. 1997. The Design of interaction. In *Beyond Calculation*. Springer, New York, 149–161.

Peter Wright and John McCarthy. 2008. Empathy and experience in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, New York, NY, USA, 637–646.

Kristie Young, Michael Regan, and M. Hammer. 2007. Driver distraction: A review of the literature. *Distracted Driving* (2007), 379–405.

Boxun Zhang, Gunnar Kreitz, Marcus Isaksson, Javier Ubillos, Guido Urdaneta, Johan A Pouwelse, and Dick Epema. 2013. Understanding user behavior in Spotify. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '13)*. 220–224.

John Zimmerman, Jodi Forlizzi, and Shelley Evenson. 2007. Research through design as a method for interaction design research in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 493–502.

Roxanne Zolin, Pamela J. Hinds, Renate Fruchter, and Raymond E. Levitt. 2004. Interpersonal trust in cross-functional, geographically distributed work: A longitudinal study. *Information and Organization* 14, 1 (2004), 1–26.