User Evaluation of OIDE: A Rapid Prototyping Platform for Multimodal Interaction

Marilyn Rose McGee-Lennon, Andrew Ramsay, David McGookin, Philip Gray

Department of Computing Science,
University of Glasgow, Glasgow G12 8QQ, UK.
{mcgeemr; adr; mcgookdk; pdg@dcs.gla.ac.uk}

ABSTRACT

The Open Interface Development Environment (OIDE) was developed as part of the OpenInterface (OI) platform, an open source framework for the rapid development of multimodal interactive systems. It allows the graphical manipulation of components stored in a structured and rich repository of modalities and interaction techniques. The platform is expected to act as a central tool for an iterative user centred design process for multimodal interactive system design. This paper presents a user study (N=16) designed to explore how the platform was used in practice by multimodal interaction designers and developers.

Participants were introduced to the features and functionality of the tool via tutorials and then engaged in an open multimodal design exercise. Participants were expected to explore various multimodal solutions to the design scenario using both traditional prototyping tools and the features available to them via the OIDE prototyping tool.

The workshops were recorded and the interaction and dialogue examined to gather feedback on how the OI tool was used or could be used to support or enhance the design stages of prototyping a multimodal application or interface. The results indicate that the OI platform could be a useful tool to support the early design stages during multimodal interaction design. The tool appeared to promote thinking about and using different modalities. The teams varied in size and composition and this appears to have an effect on how the teams approached the task and exploited the OI prototyping tool. We will offer some guidelines as to how open, rapid prototyping tools such as OIDE can be improved to better support multimodal interaction design.

ACM Classification

D.2.2 [Software Engineering]: Design Tools and Techniques. – User interfaces.

General terms

Design, Experimentation, Human Factors

Keywords

Open Interface, multimodal interaction, interaction design, rapid prototyping, user evaluations.

EICS'09, July15–17, 2009, Pittsburgh, Pennsylvania, USA.

1 INTRODUCTION

This paper presents the results from a user based evaluation of a multimodal interaction prototyping tool called the Open Interface Development Environment (OIDE). After discussing the challenge of supporting multimodal interaction design, the tool will be introduced followed by a description of the user study on how the tool was used in practice. The results will be discussed in terms of how prototyping tools such as the OIDE can best be exploited to support the early stages of multimodal interaction design.

1.1 Supporting Multimodal Interaction Design

Multimodal user interfaces (UIs) continue to grow in popularity. Alongside continued development of Graphical User Interfaces (GUIs), 3-D sound, and gesture recognition have made significant progress. While scientific understanding and empirical knowledge of multimodal interaction is increasing dramatically, our lack of understanding of how these interaction techniques can be best combined often leads to interface designs with poor usability. Although several multimodal systems have been built, their development remains a long and difficult task.

The flexibility multimodal systems offer results in the challenge of creating interfaces that incorporate multiple traditional and novel input and output modalities which can span multiple devices. Multimodal UIs are hard to prototype due to high costs in terms of the time and technical expertise required to build them. Current development tools do not address these problems appropriately. In particular, few early stage prototyping tools allow non-programmers to prototype multimodal interfaces.

In this paper we describe the OpenInterface (OI) Framework (Section 2), a component-based approach for specifying and developing multimodal interfaces intended to address these problems. Designers can graphically assemble the OI components and the code of the multimodal user interface is automatically generated. Before the OI framework is described, we briefly consider some of the existing prototyping tools.

1.2 Tools Dedicated to Multimodal Interaction Design

Tools dedicated to multimodal interaction are currently few and limited in scope. Either they address a specific technical problem or they are dedicated to specific modalities. For instance, the Georgia Tech Gesture Toolkit GT2k is designed to support gesture recognition [13]. SILK [6] is a tool for sketching graphical interfaces but does not provide multidevice prototyping. CrossWeaver [7] has extended SILK's storyboard to multimodal commands and introduces the new concept of multidevice prototyping as an extension to the work that SILK pioneered. DENIM [9] is an informal prototyping tool for web design which has sketched pages and transitions that are analogous to CrossWeaver's scenes and transitions.

QuickSet [3] is a programming platform that has been used to create multimodal applications for map and military planning and mobile domains. The QuickSet system is used for implementing multimodal applications built using the Adaptive Agent Architecture (AAA) [10]. It includes all of the capabilities for creating multimodal, multidevice applications. Prototyping by non-programmers however has not been the target of QuickSet.

The Wizard of Oz technique has been used for years to simulate interactive systems when performing both low-fidelity tests of proposed design ideas and user studies on "finished" interface designs [5]. In a standard Wizard of Oz study, a human simulates the system. SUEDE is an informal prototyping tool which exploits a Wizard of Oz design for speech interfaces [7]. All of these systems however require functioning software to be written before testing can begin.

The most effective method for constructing high quality user interfaces is an iterative approach. This requires a fast, repeated cycle of design, prototyping, and evaluation. Therefore, to be successful, a multimodal interface design tool must be easy to learn, require little programming expertise to use, and support the rapid creation, testing, and modification of interface designs. These requirements form the basis of any user interface prototyping tool targeted towards interface designers.

2 THE OPEN INTERFACE FRAMEWORK

The aim of the Open Interface Framework is to provide an open source platform for the rapid development of multimodal interactive systems as a central tool for an iterative user-centered design process. The main components of the framework are the OpenInterface Kernel, the OpenInterface Interaction Development Environment, the OpenInterface Forge and the OpenInterface Repository. Together, these tools provide a framework that can support and enhance the design and rapid prototyping of multimodal interaction.

The OpenInterface Kernel is a component based application development tool. It allows components written in common languages (Java, C/C++, Matlab, C#) to be linked into a single application. Component interfaces can be defined using an XML description language and a second description language is used to define the application structure itself.

The OpenInterface Repository is an online, editable database of OpenInterface component descriptions (see Figure 2), interaction techniques and application configurations, linked to both the OpenInterface Forge and the OIDE. It allows users to search for and download components from within the OIDE, or to find information on how to use a particular component. This is now open to the public.

The OpenInterface Interaction Development Environment (OIDE) is a graphical tool intended to aid designers and developers in working with the OpenInterface Kernel. The OIDE provides a simple drag and drop style interface (see Figure 1) to add and link components within an application, and automatically generates the XML application description required by the OpenInterface Kernel. The OpenInterface Forge serves as the central location for hosting OpenInterface software components.

At design-time, to specify the multimodal interaction of a particular interactive system under design, the designer can reuse and assemble OI components. From this assembling, the code of the interaction part of the system is generated. The open source platform therefore allows rapid development of multimodal interaction by assembling components. It will enable the rapid development of early prototypes for exploration, prototypes of different design options, and testbeds for experimental evaluation.

The OpenInterface Framework also aims to bridge the gap between academic and industrial research. The OpenInterface platform will enable the reuse of well-defined pure or combined modalities as starting building blocks. The OIDE offers a much more manageable visualization of an interface design than that offered by paper or domain independent flowchart tools such as Visio. In addition, the designs are stored in a form that, in the future, may allow them to be semi-automatically converted to fully working systems, as was done for sketched GUIs in SILK [9] for example.

From this point on in the paper the term 'OIDE' will be used to refer to the prototyping tool as a whole which includes communication to and from all the components (kernel, repository, graphical editor/OIDE) present in the Open Interface Framework.

3 USER CENTERED STUDY OF THE OI PROTOTYPING TOOL

In order to evaluate if the Open Interface Framework could be used by real designers to support and enhance multimodal interaction design, a user study was carried out. The study was undertaken with 16 multimodal interaction designers during one week of August 2008 at the University of Anon, as a series of 5 workshops (1 per day) where they were exposed to the tools and encouraged to use the tool in an open ended multimodal design exercise.

3.1 Aims

The aims of the user study were:

- to identify current design and implementation practices (including tools, methods, processes) in multimodal interaction design
- to explore how the OIDE prototyping tool is used by various potential users
- to use findings from the study to suggest how the tool could be improved

3.2 Research Questions

- What features of OIDE are used / not used / used the most / least etc.?
- How does OIDE fit into the process of interaction design?
- How does OIDE aid designers in developing multimodal interaction techniques?
- Does OIDE affect the ways in which users design multimodal interactions?
- What limitations exist in OIDE that constrain or frustrate interaction designers?

- Which unanticipated uses emerge in the work with OIDE?
- How does OIDE support existing design and development processes and techniques
- How can OIDE by improved to better fit with the conceptual models and working practices of multimodal interaction designers and developers?

3.3 Participants

16 people in 5 groups took part over the course of one week of design workshops. The design teams were allocated randomly, varied in size (N= 3, 4, 5, 2, 2), and none of the teams had worked together previously. Anyone with an interest in, or working in multimodality, multimedia or interactive systems design (industry or research) could take part. Those recruited were predominantly from a research background and ranged from Computing Science undergraduates through to PhD students to postdoctoral researchers. Participant expertise was quite varied (ubiquitous systems, information retrieval, multimodal interaction design, digital information design, web design) although all had some experience of interaction design.

3.4 Apparatus

Each group (ranging in size from 2-5) was seated around a standard PC with the Open Interface Development Environment and all other necessary OI components (as detailed in section 2) pre-installed on the machine. In addition, each team was provided with standard desktop speakers and a USB desktop microphone. A Nintendo Wii Remote controller was also provided for gesture input. Several simple digital and analog widgets (light sensor, touch sensor, slider control, LED output, force sensor) called Phidgets (www.phidgets.com) were provided with connectors and a Phidget interface board. These offered several other input and output modalities which could be used during the exercise. All of these modalities were supported fully by the OI framework and as such had equivalent components in the Repository and Forge which could be used by the participants during the exercises. Finally, participants were provided with basic traditional prototyping materials such as paper, pens, scissors, glue etc.

3.5 Design Exercise

The key design activity was an open-ended scenario-based multimodal design exercise in which the teams were given a total of 90 minutes to "*create an interactive multimodal system or a set of multimodal interaction techniques that can be used to control Google Earth as part of a museum exhibit on the Issues of Deforestation in Borneo"*.

Before users engaged freely in this design exercise, they spent the first 90 minutes being introduced to the OI prototyping environment. To achieve this, a set of three tutorials were developed in which the users were introduced to (1) the Repository which detailed the components available to them and how they should be used within the prototyping tool, (2) the graphical editor (OIDE) used to select, and connect components into an application, and (3) the set of hardware components they had available to them (Wiimote, Microphone, speakers, force sensor, temperature sensor, slider control, mini joystick control).

Using the OIDE prototyping tool and the hardware provided, teams could:

- Assemble OI components in a graphical fashion in the OIDE (see Figure 1a)
- Dynamically assemble and execute components or pipelines of components (see Figure 1b)
- Access and manipulate OI Components in the Repository (see Figure 2)
- Test multimodal input and output ideas quickly and easily in order to form design ideas and solutions for the exercise (see Figure 1 and 3)

While completion of the tutorials relied on the use of the OIDE tools, during the open ended design exercise participants were encouraged to use any methods (including using OIDE) to mock up their design.

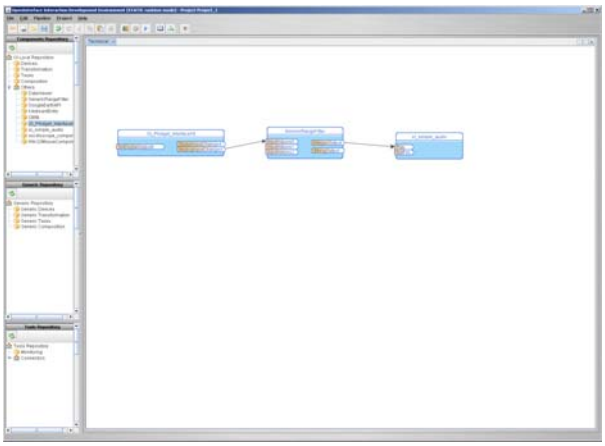

Figure 1a: The OIDE graphical interface

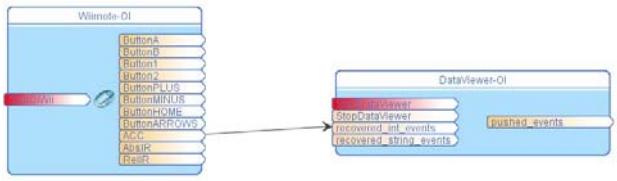

Figure 1b: Connecting components to build a pipeline

3.6 Data Gathered

Participants were video recorded and facilitators took observation notes in order that the user interaction with the tool could be recorded and analysed. Teams were also asked to produce sketches and design notes as they would normally if engaging in such a design exercise. When teams built pipelines of components using the graphical editor of the tool, screenshots of these were also saved for analysis.

Participants also invited to fill in a questionnaire before the workshop to determine what their background and expertise was as well as what their current design methods and practices were. On completion of the workshop, a second questionnaire was

filled in asking questions regarding their use of the tool and if and how they perceived it being of use to them in future multimodal interaction design practices.

The questionnaires often led to discussion and interviews structured both by the predefined questions, and by themes and issues that emerged naturally through interaction with the tool and each other during the workshop.

On completion, the following was collected across the 5 design teams:

- Video recordings (~15 hours)
- Observation notes (5/6 pages per group)
- Annotated tutorial sheets
- Paper prototypes, sketches, design notes and artefacts (See Figures 3 and 4)
- OIDE pipelines and screenshots (varied)
- Questionnaires (pre and post)

3.7 Analysis

Observation notes, videos, questionnaires and interview notes were collected and organised in the qualitative data software NVivo 8. This allowed three researchers to analyse the data for themes and issues emerging based on the Framework Analysis Approach [12]. This allowed us to examine both pre-identified issues stemming from the research questions (See 3.2), and any issues that emerged purely due to the interactions with the tool and each other during the workshop.

Field notes and video were analysed for interesting usability and user interaction issues. Pre and post questionnaires were examined to establish current practices of multimodal interaction designers and actual and potential problems with and benefits of the OIDE platform for designing multimodal interaction.

4 FINDINGS

All teams completed the tutorials and used the basic editing features and graphical editor (OIDE) of the tool (see 4.1). The most useful features were (1) the Repository, (2) the Data Viewer component, and (3) the Generic Range Filter component. These will be discussed in turn in Section 4.2.

Teams varied in their approach to the open design exercise as could be expected from such varied levels of expertise and backgrounds. All teams attempted to use the tool in the open ended exercise but did so to varying degrees. Due to the early design stage being examined, all of the teams used traditional paper prototyping tools in conjunction with the OIDE prototyping tool. The tool appeared to be most useful in enhancing communication and exploration of possible ideas, especially in the very mixed teams. This is discussed in Section 4.3.

4.1 Basic Editing Features

All basic editing operations were used successfully by the teams. These included - Add/connect/delete components, build simple predefined pipeline, build novel pipeline, start/stop components and connect input/output pins. There were some problems with deleting and then reconnecting pins on components which caused frustration for the users as there was not sufficient feedback to alert the user to the fact that the pins had not been reconnected successfully. This has since been resolved in the tool. The graphical editor tool was perceived as a useful feature for all styles of teams but especially those with more creative members. Building pipelines was seen as a useful way to connect components that they might otherwise not put together and the fact that the code was generated by the system for the designer in order to test the prototype design was seen as an attractive feature.

4.2 Thinking about different modalities

The OIDE tools appeared to help people think about alternate possibilities (e.g. different modalities) during the design phase. Being able to discuss alternative modalities in mixed teams is useful as many of the team members had expertise in one area (auditory designers for example). Having a tool which promoted exploration of alternative modalities and supported quick prototyping via the component based approach allowed those without direct expertise with certain modalities to use these modalities readily in their designs without having to consider the programming implications associated with that modality or interaction technique.

4.2.1 The Repository

The Repository (see Figure 2) was used by all groups and was perceived as being a useful tool to aid thinking about novel solutions. When teams needed to examine the Repository to determine what a component did or how it should be used this was done very successfully. In fact, many users commented afterwards that the repository was an essential feature in the system. Teams commented that they used the repository to "see what was available to them" and "get some ideas of what could be used for inputs and outputs".

4.2.2 The DataViewer

The DataViewer component was important to all of the teams. This component, when connected to an input or output component allowed users to view the raw data coming in and/or out of a component. Video analysis showed that this was used effectively (and often) as a debugging tool. It was particularly useful for testing whether components were working or switched on (they were showing data) and for exploring what type of data was being generated by a component. Overall, the DataViewer was used by all groups, provided crucial feedback about pipeline behaviour and was an essential feature of the tool rather than a separate component.

4.2.3 The Generic Range Filter

The Generic Range Filter is used to translate values from output of one component to the input of another, optionally doing some simple rule based processing. For example, the filter could produce different output if the input was an even number than if it were an odd number. This was an important feature in aiding users to determine what rules were required to send values to and from different components. Again, video analysis showed that this was a desirable feature of the tool in order to promote selecting components that might not otherwise be used together. Interviews also revealed that more technical team members expressed a desire to able to create and save collections of their own sets of rules for later use and this has since been implemented.

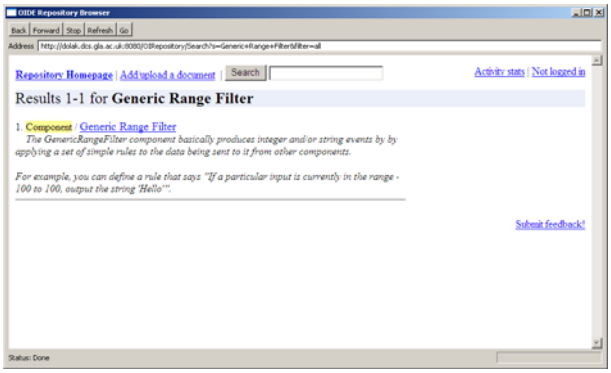

Figure 2: The component repository

4.3 Enhanced Communication between Mixed Design Teams

All teams used a combination of brainstorming, low fidelity prototyping techniques such as paper and pen sketches and the OIDE tool. The teams used traditional design artefacts to support their design process. The stages identified were (1) Brainstorming (verbal-paper), (2) Storyboarding (3) Interaction diagrams, (4) Exploration of components available, (5) Building pipelines (prototyping in OIDE) and (6) testing interaction techniques with input and output devices connected. Examples of (1-3) can be seen in Figures 3 and 4.

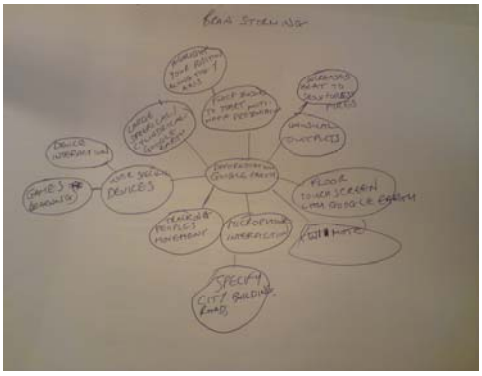

Figure 3a: Traditional mind map brainstorming sketches

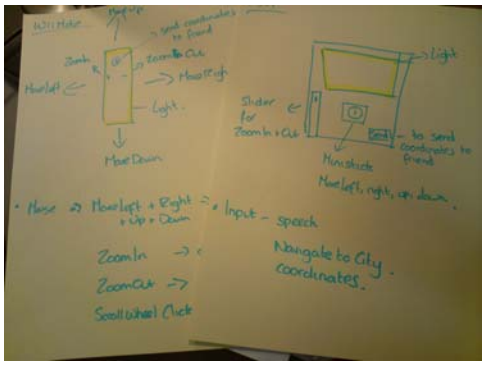

Figure 3b: Annotated interaction diagram sketches

More creative teams were not led by what was technically available and were happy to combine use of the prototyping tool with conventional pen and paper to indicate ideas that were not necessarily yet achievable with the tools they had at hand. Creative members of the team typically included non-programmers, for example graphic multimedia designers. More technical members typically included those with programming skills and/or software engineering degrees.

All teams iterated through all of these stages without prompting but as described, the emphasis and order shifted depending on both the make-up and dynamics of the team. More creative teams tended to develop full interaction ideas on paper (see Figures 3 and 4) and then implemented what they could using OIDE (they did not let OIDE limit their ideas). More technical teams on the other hand tended to form their interaction ideas based on the components available in OIDE and the physical devices and components made available to them. For this reason, technical teams commented that they would like to see new components appear in the repository as they are developed and become available.

Another interesting observation was in how the tool was used to support communication of both creative and technical ideas. Technical leads in teams used the tool to demonstrate and explain multimodal interaction techniques quickly to other team members. This allowed teams to test ideas beyond the paper prototyping stage and quickly rule in and out ideas they would continue with as a team. For these reasons, it might also be a useful 'handover' tool in teams where the design must be passed to engineers for example. This would potentially bridge the gap between ideas on paper and fully engineered solutions. It would also potentially assist industrial based design teams during handover between creative and technical teams.

Figure 4: Paper prototype generated during design exercise

How multimodal interaction design teams actually work together should be researched further in order to create tools that support the various methods and processes required more effectively. Many design and prototyping tools support technical approaches or creative approaches but don't necessarily support mixed teams. This tool could offer the support required for mixed teams during multimodal interaction design.

4.4 Improving the Tool

The evaluation served as a useful tool to identify current limitations or usability issues with the tool. These are presented here and have been resolved in the current release of the OI Framework.

It emerged from the interviews that there were not enough components yet for the tool to be used creatively in real practice. During the design exercise it was clear that for some teams there was a gulf between what people want to do (on paper) and what can be achieved in OIDE with the components available to them at present. With limited components available, OIDE at present would be more effectively used as a tool that supports collaboration and discussion when in mixed design teams. Practitioners for example might benefit more from OIDE as a tool for facilitating brainstorming, rapid prototyping, and hand over in mixed teams (as discussed in 4.3).

It also emerged that users felt that components need a more consistent design. For example some components require initialization while others do not. This caused some confusion to begin with. Users felt that components should be similar in design and component interface and that any differences in how components actually work should be hidden inside the code contained in the Forge. One solution would be to provide a look and feel guide for component writers. Components should be standardised, and work predictably.

5 CONCLUSIONS AND FUTURE WORK

Participants overall were very positive about the potential usefulness of OIDE. The most positive features according to the users were (1) the Repository, (2) the Data Viewer component and (3) the generic range filter component. The repository was an essential on-line help/reference. The DataViewer was used as a debugging tool and was an effective way to explore behaviour of devices and components. The Generic Range Filter was an important feature in aiding users to determine what rules were required to send values to and from different components. All of these features should be considered essential features in multimodal interaction prototyping tools. It was evident that these features promoted thinking about alternative multiple modalities.

Teams varied in their approach to the design exercise. All teams attempted to use the tool but did so to varying degrees. Because of the early design stage being examined, all of the teams used traditional low fidelity paper prototyping tools in conjunction with the OIDE prototyping tool. The tool appeared to be most useful in enhancing communication and exploration of possible multimodal ideas, especially in teams of mixed technically and creative ability.

As this study looked at early stage design exploration, further investigation would be necessary to determine the role and value of OIDE in other stages of interactive system development. We have demonstrated that the tool can be used successfully to support conventional brainstorming (discussion-based exploration) but that the exact use and benefits of the tool will depend on the dynamics of the group.

6 ACKNOWLEDGEMENTS

The OpenInterface Project is an IST Framework 6 STREP funded by the European Commission. The authors acknowledge the contribution of their OI colleagues and of non-OI members of the Glasgow Interactive Systems Group for acting being users in the study reported.

7 REFERENCES

1 Bailey, B.P., Konstan, J.A., and Carlis, J.V. (2001) DEMAIS: Designing Multimedia Applications with Interactive Storyboards, In Proceedings of the 9th ACM International Conference on Multimedia (Ottowa, Ontario, Canada, 2001).

2 Bouchet, J., Nigay, L., Ganille, T. (2004) ICARE Software Components for Rapidly Developing Multimodal Interfaces. Conference Proceedings of ICMI 2004, ACM Press, Penn State University - State College, October 2004, pp 251-258.

3 Cohen, P., Chen, J. Johnston, M, McGee, D., Pittman, K., and Smith, I. (1996) Quickset: A Multimodal interface for distributed interactive simulation, *Proceedings of the UIST'96 Demonstration Session*, Seattle, 1996.

4 Coutaz, J., D. Salber, E. Carraux, and N. Portolan (1996), NEIMO, a Multiworkstation Usability Lab for Observing and Analyzing Multimodal Interaction, in *Proceedings of ACM CHI 96 Conference on Human Factors in Computing Systems*. p. 402-403, 1996.

5 Dahlbäck, N., A. Jönsson, and L. Ahrenberg. (1993) Wizard of Oz Studies - Why and How. In Proceedings of *Intelligent User Interfaces '93*. pp. 193-200 1993.

6 Hong, J. and Landay, J. (2000) Satin: a toolkit for informal ink-based applications. In Proceedings of UIST'00 (November 5-8, 2000, San Diego, California) 63-72.

7 Klemmer, S.R., A.K. Sinha, J. Chen, J.A. Landay, N. Aboobaker, and A. Wang, SUEDE: A Wizard of Oz Prototyping Tool for Speech User Interfaces. The 13th Annual ACM Symposium on User Interface Software and Technology: UIST 2000. November 2000. CHI Letters, 2(2): p. 1-10.

8 Landay, J.A. and B.A. Myers. (1995) Interactive Sketching for the Early Stages of User Interface Design. In Proceedings of *Human Factors in Computing Systems: CHI '95*. Denver, CO. pp. 43-50, May 7–11 1995.

9 Lin, J., M.W. Newman, J.I. Hong, and J.A. Landay, DENIM: Finding a tighter fit between tools and practice for web site design. *CHI Letters: Human Factors in Computing Systems, CHI '2000*, 2000. **2**(1): p. 510-517.

10 Moran, L.B., A.J. Cheyer, L.E. Julia, D.L. Martin, and S. Park, (1998) Multimodal user interfaces in the open agent architecture. *Knowledge-Based Systems*, 1998. 10(5): p. 295-304.

11 Nigay, L., Coutaz, J. (1995) A Generic Platform for Addressing the Multimodal Challenge. In Proceedings of CHI'95, Katz I., Mack R., Marks L. Eds., ACM Press: New York Publ., 1995, p. 98-105.

12 Ritchie J & Spencer L (1994) Qualitative data analysis for applied policy research. In Analysing Qualitative Data, pp. 173–194 [A Bryman and RG Burgess, editors]. London: Routledge.

13 Westeyn, T. Brashear, H. Atrash, A. and Starner, T. (2003) Georgia Tech Gesture Toolkit: Supporting Experiments in Gesture Recognition, *International Conference on Perceptive and Multimodal User Interfaces*. Vancouver, B.C., November 2003.