Ollivander's Magic Wands : HCI Development

Eurydice Sophie Exintaris
Electrical & Computer Engineering Department
McGill University, Montreal
eurydice@ece.mcgill.ca

Jit Lee Lim
Electrical & Computer Engineering Department
McGill University, Montreal
jleelim@hotmail.com

Markella Georgiadis
Electrical & Computer Engineering Department
McGill University, Montreal
marky888@hotmail.com

Kyriakos Vallianatos
Electrical & Computer Engineering Department
McGill University, Montreal
kvalli@ece.mcgill.ca

ABSTRACT
This paper describes a project created to study human computer interaction (HCI) principles. The basic guidelines were to design an interactive toy which did not require a standard keyboard and screen interface. The particular project described here is a magic wand toy which can interact with both the user and with other such toys in a multi-player game. The main functionality of the toy is its ability to "cast" several distinct spells dependent on how the user waves the wand. We will describe the evolution of our prototype based on user and peer feedback and how we went about meeting our design goals.

Keywords
Human computer interaction, HCI, prototypes, interface, heuristics, user-centered design, phidget, magic, harry potter, wand

DESIGN CONCEPTS
The toy features:

- Two different types of spells (offensive and defensive) at two or three levels.
- Transparent input interface (the user simply waves the wand in the given patterns).
- Visual and auditory output.
- Multi-player interaction (offensive spells affect the target wand)

The overall initial design of our prototype was made up of several components: the interface, the mechanism detecting the waving movements, the processing unit, the physical framework, and the manual. The interface was designed so as to convey what we deemed was all the information about the state of the wand (visibility) which would be useful to the user such as whether a spell was successfully cast, what type of spell it was, whether an opponent has cast a spell at the user, and whether the wand is in a functional state. It would also have to have a minimalist design and fit the common conceptual model of a magic wand. The mechanism would have to be sensitive to various distinct movements made by the user and transmit each one in series to the processing unit. In this case we took advantage of the common conception that magic is cast using overly dramatic gestures to simplify the task. Another benefit of requiring dramatic gestures was that this also matched the common conceptual model of a magic wand. The processing unit would have to perform three distinct tasks: receive movement data, process the data to determine if it matched any of the spell patterns, and generate the appropriate output signals for the interface. The framework was chosen carefully so that it first of all looked like a wand, but also allowed us to effectively place the mechanism inside it and provide enough surface area for the interface layout. Finally the manual would also have to reflect a minimalist design and at the same time be easily understood by our target audience (children 8 years and older with a sufficient working vocabulary to understand the first Harry Potter novel).

A conceptual illustration of the toy

GOALS
At the outset, we formulated three design goals by which we wanted to evaluate our prototype. These were generally based on some of the usability design principles known as heuristics. The reader should note that these were slightly revised during the evolution of the project so as to take into account some of the design constraints we encountered. Here are the three goals with brief descriptions:

Mapping and Affordances
Our intelligent toy has the same physical appearance as a simple wand (usually perceived as a sturdy wooden branch

about 30cm long) with which most children are familiar. The mapping should therefore be straightforward and the affordances clear. This criterion can be evaluated by presenting the magic wand to users without any or with few instructions and asking them to play.

Visibility and Feedback

This criterion pertains to the visual feedback on the wand (LEDs). Is this feedback clear and visible enough? We can evaluate the visibility by showing a new user how to cast a spell without mentioning the feedback and see if she or he sees and understands whether the spell was cast successfully and what type of spell it was.

Usability

Otherwise known as the frustration measure criteria. A user should be able to cast spells and shields, within a reasonable amount of time (a few minutes after reading the manual). Also, the spell detection mechanism should be efficient enough for the user not to need to cast the spell 10 times before the system recognizes the pattern.

SOFTWARE PROTOTYPE

The goal of the prototype was to quickly test the feedback and viability of the magic wand toy. In choosing the platform to develop the software mock-up, several alternatives were considered including the use of Java, Flash, Visual Basic and C++. Ultimately the choice was to use Macromedia Flash for maximal accessibility and ease of use since most modern browsers natively support Flash. In order to provide a rich and compelling experience, we designed the prototype to be as realistic as possible while keeping the software simple: the software did not require a start/stop button to run and any unnecessary feature was removed to keep the interface clean. These efforts hopefully prevented the evaluators from getting caught up in the limitations of the software and allowed them to focus their attention to actually evaluating the magic wand toy.

The software simulates an interactive two-player game. The user can cast a defense spell or an attack spell by dragging the magic wand image in the square (see Fig.1) and forming one of five pre-defined patterns given in the manual. Ideally the software should have allowed for 3-dimensional movement detection, but faced with programming restrictions in Flash, we decided to implement a 2-dimensional simulation of the magic wand toy instead. Although we traded off the realistic aspect of the game, we gained by having a simpler interface that is easy for the user to manipulate. For evaluation purposes, we provided a manual reset button which would reset the state of both wands, a feature that would not be ported to the physical prototype.

It is noteworthy that this prototype lacks any auditory feedback. At the early stage of the design, we assumed that including sound was not of great importance because we thought it was sufficient to provide only visual feedback through the LEDs. This issue is further addressed in the Alpha system.

Feedback from Interface Mock-up and Software Prototype

The first round of feedback came from two sources: the interface mock-up and, after some improvements, from the software prototype. In both cases we were limited by the fact that a complete representation of a physical toy that needs to be moved in 3D space would be impossible to make on a computer screen. We therefore decided to postpone the testing of real user input (waving the wand) until the first physical prototype and focus instead on the mapping and visibility of the system. The one user input issue we did include was a test of how easy it was for the user to discover a spell pattern on his or her own.

The results were encouraging, but it was clear that there was much work still remaining. The testers were all able to successfully identify all the given spell patterns from the feedback the interface provided and maintained that the learning curve was short. They were happy with the way the conceptual model of a magic wand was maintained and also with how the wand was conceptually simple yet provided for interesting gameplay. From this we were at least confident that the interface and the game concept were on the right track.

The improvements that were suggested were very specific: two of the states of the wand were not visible and the lack of any sound was to the toy’s detriment and very noticeable. The inclusion of sound had been considered prior to this point, but as an optional extra feature since the visual interface seemed sufficient. Based on the feedback we realized that for a toy, sound was actually vitally important to the experience (especially since magic is supposed to produce sound), and also enhanced usability for advanced users (those who had already identified all the spells and would like to know the wand’s state without having to look at it, focusing instead on their opponents). Concerning the visibility, the testers found it confusing that we didn’t explicitly show when the wand was rendered inactive (as a result of an offensive spell from an opponent) and subsequently when it became active again. Since this negatively affected the usability and provided an incomplete conceptual model of the toy, we decided to add these states to the visual interface.

Unfortunately, the testers were not able to find the missing spell pattern. As a result we concluded that expecting the user to discover some of the spell patterns on his or her own was too complicated a task. This feature was dropped and all the spell patterns were added to the manual. Otherwise our design would fail the usability criterion since very few users would manage to cast any spells at all.

A final cosmetic change was prompted from one of our younger users. Instead of having one shield and one lightning bolt with three arcs marking the levels (as shown in the software prototype) it was suggested that children might prefer three shields and three lightning bolts instead.

ALPHA SYSTEM

The alpha system was the first attempt to develop a semi-functional system using both hardware and software. It presented a more realistic gaming experience than the first software prototype because the user is able to physically swing the magic wand toy to cast a spell which meant that the usability aspects of our toy could now be fully tested.

The framework is built using a cylindrical plastic tube with the LEDs embedded on the surface and electronics inside. The cylindrical tube provided a natural affordance for even young children since it resembles a long stick often seen in fairy tale stories and thus matches any conceptual model of a magic wand the child might already have.

While we would have preferred to use a ceramic gyroscope to accurately detect all possible movements in a 3-dimensional space, our limited budget restricted us to only rudimental objects such as glue, screws, bolts and sticks. Despite the crude tools, we were able to fabricate the toy and detect up to 4 major movements with a reasonable degree of accuracy using homemade inertial sensors (see Fig.2). The signals are fed into a computer to handle the processing of events and feedback. In several cases, loose contacts caused the Visual Basic program not to detect movements and caused the spell not to be cast.

Our original design goal required the use of infrared (IR) to communicate between multiple magic wands for multi-player game sessions. However, the cost and time to include infrared capabilities and build two magic wands were exceedingly overwhelming. Instead we simplified our design by building only one magic wand with no infrared capability and used the existing computer connected to the device to simulate the communication. The computer program, in addition to controlling the actual toy, now displays a friendly user-interface to allow users to simulate the second virtual player in a multi-player game (see Fig.3).

The physical system has 3 yellow LEDs at the outer end of the wand to provide visual feedback for the 3 levels of attack spells. In addition, it has 3 more green LEDs near the handle for the 3 levels of defense spells. Each of these LEDs would light up accordingly to the levels of attack or defense for a pre-defined amount of time. Based on the comments we received from the software prototype, we also integrated sound feedback into our system whenever a spell is cast. To overcome the difficulty of fitting a speaker into the cylindrical tube, we decided to generate the sound directly using the computer.

We included a red start button positioned next to the handle for easy access to initiate the movement detection when pressed. The computer continuously monitors the movements of the wand up until the button is released. If the movement matches any of the two pre-defined patterns, a spell is generated with the corresponding LEDs lighting up and sound playing. The addition of the start button improved the accuracy of the detector tremendously thereby leading to a higher tolerance for errors and reduced frustration for users. A small white LED is placed next to the red start button that lights up when then red button is pressed. The purpose of this LED is two-fold. First it provides visual feedback to indicate movement detection is enabled and secondly, because it should light up whenever the button is pressed, a user would know that something, such as an empty battery, is wrong with the toy.

The software presents a simplified model of the magic wand toy. No additional effort is put in to make it look realistic since its only purpose is to simulate testing. Similar to the physical device, it has two sets of three colored icons for displaying attack and defense spells. It has a drop-down selection box containing a list of spells that can be cast through the software. Spells are only cast when the user clicks on the “Cast Spell” button. A status bar provides indication if the physical device is connected. All buttons, icons, drop-down box and texts are nicely aligned and grouped in a frame according to their functions to provide an intuitive interface. In anticipation of errors, we added a debug window that shows all movements detected throughout the life of the program. This enables the evaluator to report problems that are usually deceptive or difficult to locate. The software comes with an easy step-by-step install wizard to ease deployment and installation.

Alpha System User Manual

As mentioned before, the manual was targeted to children 8 years an up with sufficient English vocabulary to understand the first Harry Potter novel. Because of the subject matter, we thought it would be nice to integrate the manual into the wizard theme as if it was a wizard manual. On the practical side, it would have to list the abilities of the wand and explain how the multi-player game works. Furthermore, based on the results of the software prototype (specifically the evaluating team’s inability to discover a spell pattern on their own), we decided to include images showing all the spell patterns available. Our final goal was to keep the manual short and the instructions simple. Doing otherwise would violate our goal of having a minimalist yet fully usable design since a system with adequate visibility and good affordances should not require lengthy instruction. All combined the manual fit into two pages including 2 images displaying the two spell patterns that were implemented at the time. The material was short enough to fit on both sides of a 12 by 18 cm card taking into account the larger font used for young readers. The final result gave information about the two spell patterns required for casting a level one defense spell and level one attack spell, the visual feedback the user could expect during gameplay, how the defense spells accumulated, and how the wand reacted when a spell was cast on it from another wand.

Feedback from Alpha System

As a first impression, the testers of our alpha system were happy that our port of the design from a software prototype to a functional physical prototype successfully kept the good functionality of our toy and its effective interface. They also found the virtual opponent an excellent compromise to our hardware restrictions because it still allowed them to use all the features of the toy.

After some use, they noticed that we had implemented only two out of the six spells but acknowledged that that this was reasonable given the limitations of the hardware components at our disposal. The fact that the defensive spells accumulated made this less of an issue because it provided a secondary means of fully testing the interface. They were concerned however that we were not meeting our usability goal since the motion recognition was very erratic and the success rate much lower than anticipated. They were able to narrow down the problem to faulty electrical contacts (with the help of the debug window provided with the software) and to the fact that the system was not very forgiving to the range of motions the spells call for. Clearly the inclusion of the button had not adequately solved the usability problem. The final interface issue the testers had something to say about was the button. They felt that it did not fit with the wand's conceptual model and wondered if it could be eliminated.

The manual was deemed to be adequate and short enough to reflect the simplicity of the interface. The testers stated that they were able to get all the information necessary from it, albeit with a little difficulty. The image explaining the attack spell was found to be confusing since it did not give any clues about its orientation. The text explaining the feedback could benefit from some restructuring and integrated into the attack and defense sections. Although the wizard-in-training passages were nice and fit in well with the toy's theme, the testers found them distracting. Finally, it was noted that the vocabulary needed to be better targeted to our audience since some slightly technical terms had been left in the text.

BETA SYSTEM

The first improvements we made to the alpha system were small, but combined they greatly improved the overall user experience and brought us much closer to our usability design goal. We replaced the faulty electrical contacts, lengthened the wires connecting the prototype to the computer by almost one meter, and made the software more flexible in the way it parsed the data generated by waving the wand. As an example, when casting a level 1 defensive spell (side to side movement) the manual indicates making a "Z" pattern. The software previously looked for a signal from the right contact followed by a signal from the left contact followed by another one from the right. In practice, however the first contact might not have been made correctly (due to the hardware limitations) so that the entire sequence of movements would have to be redone. The improved version looks for any combination of alternating left and right movements (there must be at least 3) provided that none of the other contacts are shorted. The other spells have been treated in a similar fashion. The reader should note though that the manual purposely doesn't indicate this because these modifications exist only to counterbalance the problems inherent in the hardware. As a result of these three improvements a user will have a much larger chance of successfully casting a spell. In this respect, we feel that we met our usability goal as best as we could given the time, budget, and hardware constraints.

We next concentrated on meeting our visibility and feedback goal. A final LED was added and a lock icon painted around it. This would light up when the wand became inactive as a result of an opponent's spell and turn off when the wand became active again. This provided visual feedback for the two remaining states of the wand and removed any ambiguity as to whether a wand was not functioning because of low battery power or because it was frozen. For the auditory interface, we added unique sounds for all the remaining wand states. There are now unique sounds for each attack and defense spells, for the defense level downgrading, for the wand becoming inactive, and for the wand reactivating. Since now the wand had both visual and auditory feedback for every single state, the visibility and feedback goal was met successfully.

In order to improve the playability of our toy, we also took the initiative and added a third homemade motion sensor. This permitted one more unique motion to be detected and, when used in combination with the other motions, allow for several other spell patterns. In the end we implemented a total of four patterns. Although we did not meet our goal of having three levels of each type of spell where each higher level followed naturally from the previous one, we did manage to demonstrate how the movements could be arranged sequentially to produce spell patterns of higher complexity. For example, the level two attack spell was compromised of the level one attack spell pattern, followed by the level one defense patter, followed again by the level one attack pattern. The improved usability mentioned earlier also made this more feasible.

The last improvement made to the beta system was completely painting the framework to make it look like a magic wand made out of wood. This effectively completed everything we could do with the appearance of the toy to make the mapping and affordances as clear as possible. Since it now looks like a magic wand, a child would know he or she has to wave it around to cast spells. The darker color used for the "handle" area made it even clearer where the wand needed to be held. Finally, the mapping for each LED was made explicit with the painting of all the shield, lightning bolt, and lock icons.

Beta System User Manual

The manual was improved to address all the issues mentioned in the alpha system test. The images were improved by including a human figure holding the wand which demonstrated the correct orientation for that spell. The text was better targeted to a child's vocabulary and reorganized as suggested. Finally, we made a careful choice about the wizard-in-training theme. Including it would have made the manual approach what we felt was an interesting final product. At the limit it would have been presented as a small storybook. However, in the end we decided to eliminate the theme for the most part because we wished to demonstrate how well we had met our usability goal. Presenting only the bare instructions without the theme-based clutter showed more clearly how few were actually needed. Had time and opportunity permitted, it would have been interesting to see if even some of the instructions we included were redundant by testing the beta system on some younger users.

CONCLUSION

The initial requirement of this project were to design, prototype, evaluate, implement, and refine an interactive system that does not employ a computer keyboard or a graphics display screen as part of its interface. By this very requirement, the intelligent toy that we designed and built is an example of ubiquitous computing. It does not require the user to adapt to the interface or to go out of her or his way to interact with the toy. Rather, it is the interface that adapts to the user. Indeed, the interface discretely combines itself to an otherwise inanimate wand, thus not violating the user's conceptual model of a magic wand. Ubiquitous computing is not the only human computer interaction concept that was applied in this project. The initial steps of our project design were largely influenced by Donald A. Norman's guidelines in human computer interaction [2]. More specifically, we based our design goals and evaluation criteria (mapping, affordances, visibility and feedback) on the principles set forth by Norman. Similarly, the heuristic evaluation and cognitive walkthrough performed on the software prototype in the beginning of the development of the magic wand were partly based on J. Nielsen's usability heuristics [1]. In sum, this project has enabled us to put into practice the human computer interaction principles thought in the HCI class every step of the way. As well, it has enabled us to experience first hand the inevitable tradeoffs between good HCI design and time, cost, availability or money and the difficulties of implementing an interface with good HCI principles because of the sometimes conflicting requirements.

FIGURES

Fig.1: Software prototype

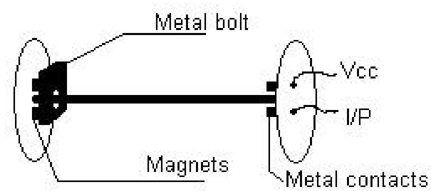

Fig. 2: Homemade inertial sensor schematic

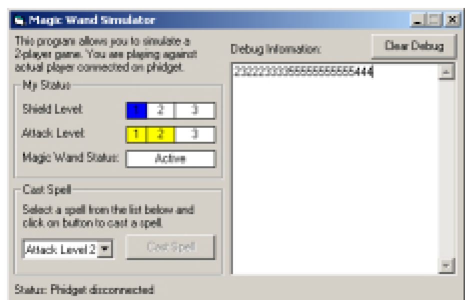

Fig. 3: Software controller interface for alpha & beta systems

ACKNOWLEDGMENTS

We would like to thank Prof. J.Cooperstock for teaching HCI, our close friends and colleagues for their great help and encouragement and our families for their patience.

REFERENCES

1. Nielsen, J. Ten Usability Heuristics. Available at http://www.useit.com/papers/heuristic/heuristic_list.html .
2. Norman, D.A. The Psychopathology of Everyday Things. New York: Basic Books, 1988.
3. Phidgets. Available at www.ece.mcgill.ca/~shassa