Low Fidelity and Usability Testing

The Introduction

ETA is simply Electronic Travel Assistance. The program runs on the Windows Pocket PC platform and serves as a digital guide for travelers venturing abroad. It not only gives the user helpful facts about destinations, but ETA also optimizes this information according to the user's needs by providing suggestions.

The purpose of these experiments is to investigate how travelers interact with ETA and how we can use the gathered information to improve our system for the user. We have chosen three typical tasks for our participants to accomplish. They are: 1. Creating the initial user's profile (Hard), 2. Finding accommodations in a desired city (medium), and 3. Locating a particular destination on a map (easy). With these three tasks, we will test the participants, gather all experimental data, and formulate a list of possible improvements on ETA.

Our Mission Statement

The group motto is simple: "create the best travel program west of the Mississippi." Many members in our group are avid travelers by nature and would like nothing more than to make the traveling experience more pleasant for everyone. However, it has been the unspoken agreement amongst the group that the current set of tools made for travelers are a bit behind the times. So being computer savvy, we have made it our personal goal to bring the world of traveling into the digital age by creating an electronic version of the travel assistance. Using cutting-edge technology like Internet connectivity and mobile hand-held devices, we hope that ETA will do to travel guidebooks like what CDs did to vinyl.

The Prototype

Our prototype is based on a revised, second version of ETA. The prototype was done as low-fi sketches on approximately 20 5" x 8.5" pieces of ordinary white paper. Buttons, pop-up windows, and the sub menu hierarchy were separate cutouts.

The main, background sketch that is prevalent throughout the low-fi prototype is the one with the main menu. Here, instead of replicating the main menu 20 times, we simply used it as a background that hovered over all the screens. The visible buttons are HOME, Maps, To Do, Profile, and Help. See Figure A.

In our first task, we asked the participant to create an initial user profile. The user accessed the profile section through the main menu. Afterwards, the participant was greeted with a series of screens that asked about her general information, budget, accommodation desires, dining, and entertainment needs. See Figure B.

As the participant went through the initial setup, she was presented with a series of available buttons that she could press and saw a couple of highlighted dialogues on the system. These cutouts made from paper were tacked onto the screen as needed to simulate the computer's reaction with the user. See Figure C.
Feedbacks in the other tasks were "faked" information screens, provided by "lists" and "maps" requested by the participant. These screens were designed to be general enough that they could be reused in other parts of the experiment but specific enough to create the illusion that the information presented was, indeed, what the participant requested. See Figure D.

The participants also had an option to choose our "help" section in case they had problems navigating ETA. This help section served as a "last recourse" that the users could take if they were truly stuck. In order to properly simulate a help section, we left it blank and just told the participant what they would have read if this feature was actually implemented. See Figure E.

The Method

Three participants were chosen for our first iteration of testing on our interface design. The participants were chosen to represent a specific profile of users for which we are developing our software and interface design. We hoped to find users that were not computer science majors, nor were they people with a strong background with computers. Furthermore, we did not want users with little or no experience at all using computers, or users that did not use technology at all as a tool for their daily routines. Our target-testing group represented people with moderate computer experience; for example, they use computers regularly for common tasks such as word processing and surfing the web. We also wanted users that have used technology such as cell phones or hand held devices, though we did not feel that ownership of such devices were necessary. In other words, we wanted users that have been exposed to technology and
are willing to use technology products, but are not experts in a computer related field. This is because design decisions that are intuitive to us as experts and developers may not be so intuitive to the average user.

Other qualities we looked for in a participant is someone who enjoys traveling, or has traveled before and has used other resources for finding information in an unknown environment. This is because we hoped that the participant could relate to the software and to the tasks we assign them, as opposed to simply following our directions word for word. This way, the participant would draw from his/her previous experiences when performing the tasks we give them. While we wanted our design to be intuitive, we did not want to radically depart from the things that users might be familiar with. We wanted users who have used other tools to find that ours fits in with their expectations. Three participants were found that met these characteristics. For the sake of confidentiality, we will call these participants Participant A, Participant B, and Participant C. A and B are both international students at UC Berkeley, and we met them both at the International House. C is a UC Berkeley student that had just returned from a long trip through Europe this past summer. Two participants were male, and one was female, so as not to skew the results because of gender.

We carried out our testing in a lounge in the International House, on Piedmont Avenue at Bancroft by the UC Berkeley campus. We chose a casual setting that would be comfortable to the participants, yet at the same time provided us the testers with adequate lighting and space to take notes and lay out our lo-fidelity prototype. Furthermore, we wanted noise and interference to be kept to minimal so that we could accurately hear the participants’ comments, but at the same time, it was not so confined as to make the user feel nervous. The user sat in a chair in front of a table on which our prototype was placed and the testers sat around the table, but far enough away so as not to distract the user.

We gave to the user the three tasks that we identified earlier in our contextual inquiry. The first, and most difficult task, was to configure the software with his/her profile so that the software can accurately provide the user with information that meets his/her travel desires and budget needs. This represented the most difficult task, the first task the user needs to do when using the software, but also the most infrequent task the user would have to do. Once entered, a user is not expected to change his/her profile very often. The second task presented to the user is to find an accommodation in the city for which the user has set up the software. We left the task as contiguous from the previous task, in that the user would start with the software in the state it was last in from the previous task. This task is of moderate difficulty, and one that the user would perform with a moderate frequency. The final task, once again continuing from the state of the previous task, is to try to find the location of a specific place. This would be the most frequent task a user in a new location would do, as we found from our contextual inquiry, and by that it should also be the easiest task.

We assigned each team member a specific role to play for the experiment. Sean was the note taker, and did not say anything during the entire experiment. Rather he simply observed the experiment and took notes for everything that occurred. Fahd greeted the participants, explained the experiment to them, and had the participants sign our consent form. He explained what our software is supposed to do, and made sure the user understood how our lo-fidelity paper cut our prototype works, as well as the rules of the experiment. For example, we would give the users tasks to perform, but we would not talk or interfere otherwise. He reassured that the purpose of the experiment is to test our design, not the user, in that we were looking for flaws with the design and not with the participant, and that the participant should think aloud as he/she does the tasks. Finally, he made sure that the participant knew we would
the tasks to the user. We made sure to keep the tasks as generic as possible in wording to allow the participant freedom with his/her choices. For example, the first task is simply given as, "You are taking a trip to Berkeley, configure the software for you trip to Berkeley." Bikas played the role of the computer, and changed the paper prototype where necessary to simulate the behavior of the computer. He did not talk during the entire experiment, and tried as much as possible to work quickly and be transparent to the participant. Once the user sat down and was introduced to the experiment, he/she was shown how the paper prototype works, so as not to be confused by the computer, Bikas, and given a brief demo as to what the computer (Bikas) will do when the participant interacts with the interface. Jun read the tasks to the user, and when the tasks were complete, we allowed to the user to conclude by going over with us any final thoughts he/she had, or any overall feelings or impressions. We allowed the participant to say whatever he/she feels about the design or the experiment. Finally, we thanked the participant for his/her time.

The goal of the experiment was to provide tests and measure for our design. On a large scale, we recorded how the user interacted with the overall layout. Namely with a home section to which the user could get to all other sections, as well as with a tool bar at the top that also links to other sections. Furthermore, a sub-toolbar would drop down where appropriate with more options for a specific section. We wanted to find out how confusing the overall layout was to the participant, as well as how difficult it was for the participant to navigate through the different sections. We wanted to know if there was a place in which the participant frequently became confused, or where he/she took a long time to figure out what to do. We wanted to know if the confusion was with the overall design, or with a specific wording choice or smaller part of the design. We had a drop down menu called to 'To Do' under which listings for accommodations, entertainment, dining, and landmarks could be found. Placing these listings under 'To Do' was a decision made based on the limited amount of space for the CE device. We wanted to measure how intuitive it was for the participant to look under this menu when trying to find accommodations. Furthermore, we wanted to know how likely was the participant to explore the menus for when trying to find where to go to find an accommodation, since we did not place the accommodation menu on the top level. In this way, we could measure how effective our hierarchy choices were. We also wanted to measure what the most intuitive means for finding information was for the participant. Did they prefer to frequently refer back to the home section, or did they use the toolbar at the top? Did they understand what our features did? Did the user understand what it meant to bookmark a listing for future reference? On a big picture level, we wanted answers to the following questions: How long did it take the user to get started on a task, or is the overall design intuitive? What sections gave the user the most problems, and why? Did the user understand what the feature we implemented does? Did the user expect something to do one thing but found it did another? How long did it take the user to recover from an error? Did the user expect to be able to do something but found he/she could not?

The Results

The tests from three participants revealed many problems with our design. For example, the participant 1, stared at our first "home" screen for a long time trying to figure out how to go about doing the first task that the participant was given. The first task was to go through the one-time set up of a traveler profile and it was meant to be a 'hard task'. She seemed to be wondering what the "arrow" that is pointing at PROFILE button symbolized; arrow was drawn to
shift user's attention to the PROFILE button. In fact, other two participants had hard time pressing that PROFILE button to start up the process. They tried clicking on the USER INFO button from the top menu. Consequently, he attempted to enter his DINING information by clicking DINING button, which was intended for other purposes. When the participant finally clicked on the PROFILE button, they needed to press CREATE PROFILE button and it asks for the destination address. However, the majority of our participants entered their own address into that address textbox.

Then, the participants were asked to enter the duration of their trip, some of them mentioned that they want to leave in a month and stay there for a week but did not know the today's date, talking in terms of duration instead of absolute date. Then, the users were asked to enter the budget for the trip. This part confused two of the three participants because the interface did not explicitly mention whether the budget amount is for the whole trip or for just for one day during the trip. One of the participants tried to calculate the total cost on his own, whereas we meant it for the program to calculate it for the travelers automatically. Then, the next screen asked travelers to enter their food and entertainment preferences from the menu. One of the participants asked for the type of food that would be permitted by his religion. When we asked them later, none of the users knew that they could have picked multiple items from these menus.

Once, the first task was completed, we had our participants find accommodations from the existing screen which is the last screen from the previous task. In doing so, majority of the participants clicked on the wrong button. For example, the user needed to get out of the current screen completely and pick a new menu under To Do, which contains several of task related functions. All of the participants failed to click on To Do in order to find accommodation. Instead, they clicked on the existing sub-menu "accommodation" that was part of our setting up accommodation process from the first task. All of our participants ended up clicking on the "?" button in order to find out, the accommodation was grouped under To Do. When the participants were presented with a list of available hotels, they clicked on one of the hotels successfully. Then, they were presented with the map of the particular hotel along with some details and a BOOKMARK button. Some participants wondered if one could make a reservation by clicking on the bookmark button and the others did not know why one had to bookmark the hotel. Also, one of the participants complained that once the user choose a particular hotel, one has to bookmark the hotel because there is no BACK button and it would be impossible to go back without bookmarking it.

Lastly, we tested users with the function of using a map and it was meant to be an EASY task. And, not surprisingly, all of the participants had no trouble using it. However, they mentioned that they would like to see the bookmarked hotels on the map and they would like to be able to click on some point on the map to zoom-in. In addition, they would like to see a list of tourist attractions and other interesting places in the vicinity and have it marked on the map when selected.

Our Discussion

From our tests, there was no "fatal" error that we overlooked from our LOW-FI design. However, there were many errors that we need to consider more carefully. The most important problem that we found problem with was To Do button. The button was created in order to group tasks based on their action such as finding an accommodation. However, when the participants were presented with a task that involved making an accommodation, not one of the participants was able to click on To Do and click on appropriate menu from its submenus. They
complained that To Do sounds like a section where they keep a list of things to do while visiting a place, such as "see the Eiffel Tower." We thought that we could make To Do button into separate buttons and introduce the side scrolling arrows to view other menus that are not currently viewable.

During the profile wizard, some participants were confused about entering in the budget information because they were not sure whether they needed to put in daily budget or total budget for the whole trip. In addition, they were not sure that they needed to enter in their address or destination address in the Address box. Similarly, some participants were confused about the bookmark button in accommodation. We need to be more descriptive in the instructions without being too wordy and clearly explain what each step involves and what each button does without leaving the current screen with a popup help menu.

There was also a problem with the detailed hotel information because the page does not have a back button; the users are forced to bookmark the hotel once looking up the detailed info. In addition, the participants did not know that they could pick menu items from the list of 'type of food', 'type of preferred music', etc. We thought that we could turn the menu into check box in order to compensate for this confusion. Our participants thought that the map would not be as useful if it did not have a list of popular destinations on it along with the hotels that were bookmarked. That way the would be able to see how far each places is relative to the place they are staying at.

Even though we got a lot of feedback from our participants, there are many aspects we could not measure with these tests. For example, we could not measure how efficient the design is for the expert users who are very familiar with our program, ETA, and want to perform a task as fast as possible. In addition, there are many other features that we could not test because of time constraint with each user.

Note: Demo Script is included as part of our Methods Section, so you can refer to it there.