1.) Thinking About User Interface Design

Interface #1: Alarm Clock – In the morning I am always greeted by this device. I immediately put my hand over it and press the biggest button on it, “snooze.” Most people hit this button when they are half awake, so knowing this the designers should always make the snooze button the biggest button so that it is easy to find and be pressed. If the user has to go through a lot of effort to snooze, then the user ends up waking up and that defeats the purpose of being able to be half asleep for a little longer. Another consideration that should have gone into the design is making the screen an LED light so that the time can be visible when the user wakes up in the middle of the night and is wondering what time it is. The last strength of the interface in my alarm clock is that it has a bell on the screen to signify that the alarm is on. This makes it very easy to know whether I have set the alarm or not. Some of the weaknesses in the clock are that the alarm sound is not customizable. I cannot pick how loud I want the volume to be, or different tones. Right now the volume is REALLY loud. I believe I could easily wake up with something that wasn’t nearly as loud. I also can’t determine how long the snooze frequency is. Right now it snoozes after every 4 minutes. I also don’t like how there is a small switch on the top to change between “set alarm” and “change time.” I often change the time when I actually intended to set the alarm. This is because the switch is really small and it is difficult to tell whether it is on set the alarm or change time. Lastly, I don’t like how I can’t preset more than one alarm at once. On Mondays, Wednesdays and Fridays I need to wake up at a certain time, and on Tuesdays and Thursdays at a different time. If I could program my alarm clock to always go off at 9am on Mon/Wed/Fri and 10am on Tues/Thurs then I would never have to worry about forgetting to set my alarm in the morning. With a redesign, the alarm clock could become more customizable, have bigger buttons to change between “change time” and “set alarm” and have more elaborate features, such as setting alarms for different days of the week.

Interface #2: Cell Phone – I use my cell phone a lot. I like how I am able to customize features to be available when I push the up/down/left/right arrows. For example, I use the calendar feature a lot, so now whenever I open my phone and push the up arrow, my calendar is displayed. Another feature I like is how the menu is customizable. I can pick between a horizontal menu with words and a square menu with pictures. I can also pick different shades of colors to have for the font, and a picture for my screen display. I like this because it makes my phone seem unique, even though many other people also have the same phone. The last thing I like about it is the outer screen, which displays the time without me having to open my phone. It also has a menu strip that comes up with different pictures to represent missed call/text message/voice mail, etc. This way I can know if I missed a call or something by discretely taking my phone out of my pocket. One thing I don’t like about my flip phone is that I can’t put it on silent or turn it off when it is closed. In order to change the ring volume I have to open the phone. In other words if I’m at a business event and I forget to put my phone on silent, I can’t just reach
in my pocket and do it discretely. I also don’t like how I can’t mute the camera. Every time I take a picture with it, there’s a really loud noise that goes off. I might not want everyone around me to know that I’m taking a picture, but my cell phone insists on making sure that the whole world knows. Lastly, I don’t like how whenever I am navigating through a menu I have to push “clear” to go back a menu. On my old phone, there where words “Back” and “Select” on the bottom left and bottom right of the screen, directly above two buttons on the phone. This made it really easy to know how to go back in the menu, without having to click on “clear.”

Interface #3: Camera – I recently bought a Canon digital camera. There are many features that I like and many that I don’t like. A feature I like is a button with the picture signifying “flash.” I’ve noticed that in a lot of cameras you have to go to the menu and somehow find the flash. It seems to be a commonly used feature, worthy of its own button and it has it in my camera. Another feature I like is the picture of a battery with 3 bars that displays how much power my battery has left. This is always visible, so I always know when I should charge my battery. The third feature I like is the zoom switch. It is located around the button that you push down to take a picture. If you move the switch to the right, it zooms in, and if you move it to the left, it zooms out. It is conveniently located right by the button to take a picture, and to me, right signifying zoom in and left signifying zoom out is pretty intuitive. A feature that I don’t like is deleting a picture. When viewing a picture, if I push the down arrow, then a menu pops up asking if I want to erase that picture or cancel. If I hadn’t accidently stumbled on that, I never would’ve known how to easily delete a picture (the long way is by going to menu->options->delete->delete current). Another thing I don’t like about this is how it doesn’t ask me if I’m sure that I want to delete that picture. In other words, if I accidently push “down->ok” then my picture is deleted. Most cameras seem to ask you if you’re sure you want to delete something before actually deleting it. Lastly, I don’t like how it doesn’t tell me how much space I have left on my memory card. This past summer I was on vacation in Brazil and ran out of memory space in the beginning of a tour and I hadn’t brought an extra memory card. If my camera could somehow display how much memory space there was left on my memory card, I would’ve been prepared and would’ve brought and extra card.
2.) Learning to Listen to Users

Three people were interviewed. I will refer to them as James, Phil and Judy.

James is an experienced desktop computer user who works full-time in a technical position. In his job James is forced to use Microsoft Outlook for his outgoing and incoming emails. He is often frustrated by how many different options the program has, when all he really cares about are 2 features: sending/receiving emails and the calendar. James believes that Outlook has many unnecessary features that “get in the way” and end up making the interface too messy and “scary looking.” James believes that it is “impossible to take in all the information on the screen.” He thinks that the design could be greatly improved if Outlook conducted some type of study to determine how often certain features are used, and hid the ones that aren’t used as often so they aren’t taking up so much room in the display. At work James is also forced to use Internet Explorer 6. What mainly frustrates him about this application is the fact that one window cannot open multiple tabs. Therefore whenever he wants to access multiple websites, he has to open several different windows. James wishes his company allowed him to download Internet Explorer 7, which does contain a multi-tab feature. James also has a problem with Firefox. He believes that the “Tools -> Options -> Advanced” selection is ridiculous. This is because the “Advanced” tab has 4 other tabs that lead to even more information. This can be very frustrating for a user, especially since the words “Options->Advanced” aren’t very descriptive. James believes this could be fixed if there were more explicit titles for the options and if the 4 subcategories of Advanced were perhaps categories of their own (Network, Update, Encryption and Main).

Phil is a current Computer Science student at UIUC. He says that what frustrates him the most are programs that don’t need to be running on his desktop and take up a lot of memory on his computer. The process that are running probably need to be running for some reason, but Phil doesn’t realize this. Maybe the processes could have more explicit names to imply why they are needed. Phil is a Microsoft Zune owner. He doesn’t like the Zune software interface. He said it’s slow and there is no drag and drop feature when creating a new playlist. Because of this a lot of effort is involved when syncing music. The software should add a drag and drop feature to provide a better experience for users trying to sync their music. He also mentioned that after installing the software, you have to wait several hours before being able to use it, which is frustrating. Lastly, Phil, like most UIUC students, struggles a lot with Compass. Compass is a website where several professors post class assignments and coursework. Compass always takes a really long time to load and freezes every process when it is trying to run. Phil believes this is because it is attempting to start its Java applet, when according to him, “No one even uses the stupid Java applet for chatting on Compass. They should just load that if you decide to use the chat.”

Judy is also a Computer Science student at UIUC. She mentioned being frustrated with Microsoft Office 2008. She believes that the options up at the top have too many features on them, reducing the amount of space. She has a small screen on her laptop, so to her, the extra space would be
convenient. She said she’s heard that there is a way to minimize the table, but she still hasn’t discovered how to do it. The design would be a lot better if there was an obvious way of minimizing the table. She is also frustrated by the new Facebook layout. She understands that sometimes companies want to revolutionize and come up with a new design for their products, but often times she wishes they wouldn’t do that. With the old Facebook she already knew where to find everything and it was easy to navigate through. Now she claims that the buttons are all hidden and that it took her forever to find out how to change her picture. After a long time of trying to figure it out, she accidentally hovered the mouse over her profile picture and an option came up “change picture.” She says that before it was under “Profile->Edit” which is a logical location for changing your profile picture. Lastly, Judy is frustrated by emacs. She sometimes uses it for programming in her classes. She says that it takes a really long time to cut and paste if you use the “Edit->Cut” option, but is really fast if you know the keyboard shortcut. This doesn’t make any sense. She is also frustrated by how it isn’t customizable, so she can’t change the color of her screen or the color of the font. Her frustrations could be diminished if the keyboard shortcuts were made more visible and if there were more ways to easily customize the screen.
3.) Power Law of Practice

<table>
<thead>
<tr>
<th>Trial</th>
<th>Time*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>35.657</td>
</tr>
<tr>
<td>2</td>
<td>13.968</td>
</tr>
<tr>
<td>3</td>
<td>17.328</td>
</tr>
<tr>
<td>4</td>
<td>16.032</td>
</tr>
<tr>
<td>5</td>
<td>11.734</td>
</tr>
<tr>
<td>6</td>
<td>9.656</td>
</tr>
<tr>
<td>7</td>
<td>9.141</td>
</tr>
<tr>
<td>8</td>
<td>14.969</td>
</tr>
<tr>
<td>9</td>
<td>14.016</td>
</tr>
<tr>
<td>10</td>
<td>8.219</td>
</tr>
<tr>
<td>11</td>
<td>13.484</td>
</tr>
<tr>
<td>12</td>
<td>6.734</td>
</tr>
<tr>
<td>13</td>
<td>7.547</td>
</tr>
<tr>
<td>14</td>
<td>5.14</td>
</tr>
<tr>
<td>15</td>
<td>6.703</td>
</tr>
<tr>
<td>16</td>
<td>6.656</td>
</tr>
<tr>
<td>17</td>
<td>8.907</td>
</tr>
<tr>
<td>18</td>
<td>3.891</td>
</tr>
<tr>
<td>19</td>
<td>9.062</td>
</tr>
<tr>
<td>20</td>
<td>6.875</td>
</tr>
<tr>
<td>21</td>
<td>4</td>
</tr>
<tr>
<td>22</td>
<td>4.688</td>
</tr>
<tr>
<td>23</td>
<td>9.813</td>
</tr>
<tr>
<td>24</td>
<td>4.125</td>
</tr>
<tr>
<td>25</td>
<td>4.422</td>
</tr>
</tbody>
</table>

* red = incorrect  
green = correct
Graph:

Analysis:

\[ T_n = t_1 \times n^a \]

For this graph, we have \( y = 30.82x^{0.54} \).

So \( a = 0.54 \)

Strategy:

At first, subject had to recite the alphabet from the beginning to try to remember what the last 8 digits were. Interestingly, towards the end the subject memorized the strings “zyxw” and “vuts” instead of 1 single string “zyxwvuts.” By memorizing this, the subject could easily type the last 8 digits instead of having to think about it as hard as in the beginning.
#include <iostream>
#include <fstream>
#include <string>
#include <cstdio>
#include <ctime>
using namespace std;

int main()
{
    string inputString;
    string correctString = "zyxwvuts";
    int resultsCorrect[25];
    float resultsTime[25];
    int counter;

    cout << "Enter the last 8 digits of the alphabet backwards: ";

    time_t start_time;
    while (counter < 25)
    {
        start_time = clock();

        if (getline(std::cin, inputString))
        {
            float time1 = (float) (clock() - start_time) /
                           CLOCKS_PER_SEC;
            resultsTime[counter] = time1;

            if (inputString.compare(correctString)) //incorrect
                resultsCorrect[counter] = 0;
            else
                resultsCorrect[counter] = 1; //correct

            system("CLS");
        }
        counter++;
    }

    int i = 0;
    while (i < 25)
    {
        cout << i << " " << resultsCorrect[i] << " " << resultsTime[i] << endl;
        i++;
    }

    return 0;
}
4.) Choice Reaction Time

(a) \[ MT = 545 + 420 \times ID = 545 + 420 \times \log_2(7) = 1724\text{ms} \]

\[ ID = \log_2(A/W + 1) = \log_2(6+1) \]

A/W is 6. Say you start on the top of the menu. Then, on average it takes 6 units (size of a button) to move the mouse to any of the 12 buttons below the top button.

(b) 4 items in dynamic menu:

A/W is 1. Say you start on the top of the menu. Then, on average it takes 2 units (size of a button) to move the mouse to any of the 3 buttons below the top button.

Hence \[ MT = 545 + 420 \times \log_2(2+1) = 1211\text{ms} \]

8 items in static menu:

A/W is 8. Say you start on the top of the menu. Then, on average it takes 8 units (size of a button) to move the mouse to any of the 7 buttons below the top button.

Hence \[ MT = 545 + 420 \times \log_2(8+1) = 1876\text{ms} \]

Case 50/50: chance of either menu, so \[ MT_{\text{TOTAL}} = \frac{(1211 + 1876)}{2} = 1543\text{ms} \]

Case 75/25: \[ MT_{\text{TOTAL}} = (3/4)(1211)+(1/4)(1876) = 1377\text{ms} \]

Case 90/10: \[ MT_{\text{TOTAL}} = (9/10)(1211)+(1/10)(1876) = 1277\text{ms} \]

The minimum choice time is when you only have to select from the dynamic menu, yielding \[ MT=1211\text{ms}. \]

The probability split that would cause maximal choice time is 0/100. This would mean that you would be picking only from the static menu and not from the dynamic menu, which is in the top of the menu.

(c) An important limitation of Hick’s law is it doesn’t take into account “thinking time.” For example, if a user is on the show “Who wants to be a millionaire” and asked to answer a question on a computer. He/she might take some time to think about the answer to the question, hence not answering right away. Hick’s law would not take this into account.