CS 465
Homework 2

[Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.]

Albert Albert [ albert2@illinois.edu ]
9/10/2008
Problem 1

3 non-desktop user interfaces:

- Mobile Devices
- Portable GPS (Mio, Garmin, etc)
- Digital Camera Pocket size

Different Social/cultural/work contexts in which the interface is typically used.

Mobile device: used very often, this device sometimes used for calling emergency service. Therefore, the design for this device should be portable, intuitive, and uniform interface for all brands (press number buttons to input the number and press the green button to dial). The uniform interface is for mobile device is the good design.

Portable GPS: moderate use, this device will need simple user interface. The menu depth should be 3-4 at maximum and will have automatic locking mechanism to prevent user to operate the device when driving. The way portable GPS designed today is bad in my opinion (more to be discussed at the weaknesses section)

Pocket Size Digital Camera: moderate use, this device should only served as a device for taking pictures and produce good pictures without any configuration from the menu. Therefore usually, the camera maker create automatic feature on the camera. The design for pocket size digital camera already perfect, it can fit in your pocket and have long battery life.

Three Strengths

Mobile Device:

1. Pocket Size: Since the device is “mobile” size is matter. The designer did a good job on designing the mobile device interface, since all mobile devices can be fit in the pocket.
2. Uniform Interface (Number Button + Green and Red Button): the decision to make mobile phones have uniform interface is good part for this device. Even for the new user, using the basic function (calling people) can be done intuitively on all phones.
3. Speakerphone mode: This allowing user to have some kind of conference call and also eliminating car driver to answer phone while driving which is the good part of the design.

Portable GPS:

1. Touch screen interface: the designer did a good job in eliminating button confusion by minimizing button number to 1 (on/off button)
2. Pocket Size: portable GPS mean that the GPS can fit in pocket, the designer also did a really good job.
3. Voice guidance: The designer really did a good job in adding the voice guidance feature, driver will not lose his eye focus by looking to the GPS screen.

Pocket Size Digital Camera:

1. Automatic mode: this feature allowing first-timer to directly used the camera which is a good feature for all camera (abstraction to the all buttons).
2. Uniform interface: shoot button always on the same place is one example that is a good interface in this device.
3. Image Stabilizer: this feature is very well designed because it involves biological research, human hand by nature cannot 100% not moving there will be small vibration.

**Three weaknesses**

Mobile Device:

1. Each brand has their own accessories (chargers): they use different chargers for all different brands, uniform charges (usb chargers) can be the solution to solve this problem.
2. Different menu interfaces (messaging): each brand also have different way to access the messaging feature, one solution is to create uniform solution (like voicemail button always pressing number 1).
3. Should have automatic “silent” mode: as a human, we sometimes forgot to silent our mobile phone. Since all mobile phones have the calendar feature, the mode should be corresponds to the calendar. For instance, if there is a meeting in the calendar, the device should automatically in the silent mode.

Portable GPS:

1. No Voice command: Since the GPS have voice guidance, it should have the voice command also. This design will eliminate all the interaction by hands, thus reducing the accident rate by operating device while driving.
2. Sun light make the screen hard to read: the interface design should include something that can make screen always visible in all conditions.
3. Bad Interface (screen direction doesn’t reflect real world): With the today’s technology it’s very easy to add 3D model in the GPS system to reflect the real world, this 3D interface will be the solution to make GPS even more user friendly.

Pocket Size Digital Camera:

1. Complex configuration to produce good pictures: Automatic feature that is not provided is not always producing good pictures; the designer should have automatic features that always produce good pictures.
2. Not very useful features usually added (mp3, video): Camera is for taking pictures, addition for multiple features causing the device not optimal for the main features. The redesign should only use camera as a device to take pictures not as a multi-device.
3. Screen doesn’t reflect real images: The designer should design the screen to reflect the real images without any improvement. The camera screen usually not real pictures that will be displayed in the monitors, camera screens tend to improve the quality of the images. The redesign should make the screens in the digital camera to reflect real images.
Problem 2

Microsoft Paint

Problem they encounter:

- Resize Images
- Difficulty to edit the text when the text box is gone
- Save Image from “Print Screen” involves lot of steps

How frustrating?

He was very frustrated, because he doesn’t want to buy special software for doing a simple function such as resizing images, creating simple brochure, or get the images from memory buffer and save it as a file.

How much these problems affect their productivity?

It affect their productivity, because every time he want to do basic things such as resizing images, he need to send the file to people that has software such as Microsoft Office Picture Viewer to resize the images for them. Usually this way is not very time efficient and therefore reducing productivity.

What could be done to correct them?

Add-on feature to Paint to perform simple and easy image resize.

Analysis:

Based on my observation, the product developer want to promote another product that can do more function by giving the user very limited software. My design that can solve the problem is by adding the basic feature such as image resize and the others to the existing product.

Microsoft Calculator

Problem they encounter:

- Cannot stores number
- No sample usage for scientific feature
- Bad Interface, they can write $x^2$ but they write $x^\text{2}$

How frustrating?

The first problem about storing number especially is very frustrating, they will need to have notepad open for storing temporary values.
How much these problems affect their productivity?

This problem very affecting productivity because calculation accuracy is reduced because storing more than 2 values in notepad without any label will be confusing.

What could be done to correct them?

Change the interface to look like real calculator and have button to stores value for the scientific feature, because scientific calculator will always have storing number button.

Analysis:

Based on my observation, the software created by the software publisher is not updated since Windows 95. My guess is they want to develop something that can be solved using real calculator more easily. Another way to resolve this issue is by actually implementing all of the function.

Windows Vista

Problem they encounter:

- Very Slow boot up
- Many not useful features such as Aero that consumes lot of memory.
- Lot of menu structure changes from its predecessors.

How frustrating?

For a low budget computer, it’s very frustrating to use new operating system like Vista

How much these problems affect their productivity?

This problem is very affecting productivity because by using lot of memories that means the computer will easily crashed when all of the available resources is used.

What could be done to correct them?

Remove the not useful features to free up some memory.

Analysis:

Based on my observation, this is very true since vista eats up 50% of the available RAM instead of Windows XP that only eats up 30-40% of the memory. 10% difference for the memory usage is used by the Aero features that I never used. My solution to this problem is to disable this feature.
## Problem 3

<table>
<thead>
<tr>
<th>Trial</th>
<th>Correctness</th>
<th>Time (In Milliseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TRUE</td>
<td>3482</td>
</tr>
<tr>
<td>2</td>
<td>TRUE</td>
<td>3551</td>
</tr>
<tr>
<td>3</td>
<td>TRUE</td>
<td>2733</td>
</tr>
<tr>
<td>4</td>
<td>TRUE</td>
<td>5646</td>
</tr>
<tr>
<td>5</td>
<td>TRUE</td>
<td>2447</td>
</tr>
<tr>
<td>6</td>
<td>TRUE</td>
<td>3391</td>
</tr>
<tr>
<td>7</td>
<td>TRUE</td>
<td>2158</td>
</tr>
<tr>
<td>8</td>
<td>TRUE</td>
<td>2270</td>
</tr>
<tr>
<td>9</td>
<td>TRUE</td>
<td>2350</td>
</tr>
<tr>
<td>10</td>
<td>TRUE</td>
<td>1951</td>
</tr>
<tr>
<td>11</td>
<td>TRUE</td>
<td>2174</td>
</tr>
<tr>
<td>12</td>
<td>TRUE</td>
<td>2206</td>
</tr>
<tr>
<td>13</td>
<td>TRUE</td>
<td>3099</td>
</tr>
<tr>
<td>14</td>
<td>TRUE</td>
<td>2319</td>
</tr>
<tr>
<td>15</td>
<td>TRUE</td>
<td>2543</td>
</tr>
<tr>
<td>16</td>
<td>TRUE</td>
<td>1886</td>
</tr>
<tr>
<td>17</td>
<td>TRUE</td>
<td>2254</td>
</tr>
<tr>
<td>18</td>
<td>TRUE</td>
<td>1999</td>
</tr>
<tr>
<td>19</td>
<td>TRUE</td>
<td>1806</td>
</tr>
<tr>
<td>20</td>
<td>TRUE</td>
<td>2062</td>
</tr>
<tr>
<td>21</td>
<td>TRUE</td>
<td>2030</td>
</tr>
</tbody>
</table>
Learning constant calculation:

Trial 4th = 5046 ms (T1)

Trial 21st = 2030 ms (Tn)

α = 0.2945802495128714054725415261516 = 0.294

Explanation: I got the value α from formula given in the lecture notes Tn = T1 * n^(- α), I choose the point from the 4th and 21st trials because it refers to the worst and best time based on the subject.

Comment on subject strategies:

Starting from trial 7th the subject refers the last 4 letters as the word “vuts” making him to type faster.

Source Code (.java attached) Algorithm below

```java
import java.util.*;
import java.io.*;

public class problem3
{
    public problem3 ()
    {
        System.out.println( "Hello World" );
    }

    public static void main ( String[] args )
    {
        BufferedReader br = new BufferedReader( new InputStreamReader( System.in ) );
        String userInput = new String();
        Calendar cal = Calendar.getInstance();

        try
        {
            long startTime = Calendar.getInstance().getTimeInMillis();
            userInput = br.readLine();
            if ( userInput.equals("zyxwvuts") == true )
            {
                long endTime = Calendar.getInstance().getTimeInMillis();
                System.out.println("Correct");
                System.out.println("Time = " + (endTime - startTime) + " milliseconds");
            }
            else
            {
                long endTime = Calendar.getInstance().getTimeInMillis();
                System.out.println("Wrong");
                System.out.println("Time = " + (endTime - startTime) + " milliseconds");
            }
        }
        catch (IOException e)
        {
            System.out.println( "something wrong with IO reader" );
            System.exit(1);
        }
    }
}
```
**Problem 4**

a. Time for the user to select an item from the menu will be

Using Fitt’s law: \( C = B \log \left( \frac{S}{N} + 1 \right) \)

Assumption:

- Mouse movement take \( \frac{3}{8} \) screen size on average
- Menu size will approximately \( \frac{1}{10} \) screen size if it contains 12 items

Based on Fitt’s law the response time will be \( C = B \log \left( \frac{\frac{3}{8}}{\frac{1}{10}} + 1 \right) \)

b. Time for the user to select an item from the menu will be

Dynamic/static

50/50

Time to access item that located in the dynamic area will be lower than the static one because the dynamic one has greater area (4 items)

75/25

Time to access the item that was located in the dynamic area even greater than the static area’s menu. Menu in the static area will be very difficult to access since it will be very pack.

90/10

It will be almost impossible to access the static area menu if the only got 10% from the menu area.

*Minimum choice time for the dynamic menu*

Single hit.

*The probability split that would cause maximum choice time*

Dynamic Menu: 4 Items (4/12 x 100% = 33.33% total area)

Static Menu: 8 Items (8/12 x 100% = 66.67% total area)

c. Limitations for the Hick’s law is it involves lots of abstraction in predicting real world performance tasks. Lot of variable is neglected and therefore it will be very hard to compute exact number calculation for solving the real world problems.