CS465 HWK#5
Empirical Study

• **A brief overview of each interface used in the study. Include URLs if appropriate.**
  
  List based HelperFinder (netfiles.uiuc.edu/cmin4/cs465): The list based HelperFinder is to help a user to find matched person on their classes. The way to navigate to have a matched person is displayed as a list. Each event will correspond to another list with relative information.

  Map based HelperFinder (localhost): The map based HelperFinder is to help a user to find matched person on their classes within the map. Almost all the matching options can be alternated within the map without switching to other screen.

• **The narrative for each task that you gave to users.**
  
  Before the experiment: Imagine you have classes you want to get help and classes you can help. Would you want to use your strength in certain classes to get help? HelperFinder will easily let you do so. You can assume that all the data is already present and All you need to do is to find a person who is highly matched with you and contact the person. Also you need to be able to update your classes you can help.

  For matching: Assume that you have classes you can help and classes you want to get help. You need to find the highly matched person among other users. All you need to do is to navigate through the menu and get to the point where you will see a list of people sort in how well you are matched. Once you find a person you think you will like to contact, you need to send a message to him. The person wont know any personal information about you so don’t worry.

• **The script you read from at the beginning of the study.**
  
  Attached.

• **Questionnaire forms.**
  
  How many classes do you usually take in one semester?
  1 2 3 4 5 more

  Do you feel the need of peer tutoring on certain subject?
  [Never] [Sometimes] [Very often]

  Would you be willing to help others if they are to help you on the subjects you want to help?
  [No][Maybe][Yes]

• **Description of the metrics used and an explanation of why they were selected.**
Goals of the experiment: delete OFFER CLASS from personal info, confirm the updated OFFER CLASS, send message to a matched person.

What error means in this experiment: There is an optimized way for a user to achieve his/her goal. If a user is taken the wrong path to achieve the goal, it is defined as error. I.E) click [INFO] instead of [MESSAGE] when asked to a user needs to send a message.

Frustration level: Vocal/non-vocal expression of a user during the experiment especially when they are being stuck at IDK state (i.e. "i don't know what to do to send a message") for each subject of the experiment

   1. Completion time: Total time it takes the user to accomplish the goal of the experiment
   2. Click performed to achieve the goal of this experiment.
   3. Error recovery time
   4. Frustration level

• A summary of the raw data collected (spreadsheets, interaction videos, etc.). Attached.
• Visual summaries of the data used to make sense of the results. Attached.
• Interpretation of the results.

   By using both the list and map interfaces, we were able to make several conclusions on reactions and the level of difficulties on each interfaces. Users did seem to take more time to achieve given tasks on the list interfaces than map interfaces. On the list interface, users seemed to focus more on the mapping of the button provided with the task they were asked to do than achieving the goal. Users also seemed to have hard time locating the proper buttons to initiate tasks as well as trying to figure out what went wrong once they were in the incorrect state of the application. On the map interface, users seemed to do required tasks faster with less errors as all the related sub-tasks were displayed within the screen. Occasionally, however, there were more clicks performed when users tried to recover from mistakes they made

• Recommendations on how to improve the interfaces

   During the experiment, we were able to find few things that needed be improved. One of which is a better indication label of what buttons do. People were frustrated when they were not able to figure out what to do to recover from mistakes they made. For instance, we used [HELP] to indicate kind of classes a user can help, but it was many times interpreted as "Help" which one can find from their OS.
#Attachment 1: The script from at the beginning of the study

**HelperFinder**  
Comparing Interfaces

**Objectives**  
This study is being done to compare two different types of interfaces on the same system. What they are going to accomplish is exactly the same but how it is done is slightly different. In this study we are trying to find which is better than the other and what kind of improvements can be made on each of the interface. Our application is to find the best helper around the area so that you can find helpers to help you with the subject you struggle with. Currently we are assuming that the helps needed are the class works only and nothing outside of the class.

**Special equipment**  
No special equipment is used; however, computer is used for one of the interface. The other interface is done with the paper prototype.

**What data is being collected**  
Several data are going to be collected to determine which interface is better. First of all, the overall time it took to accomplish each task will be recorded. The time it takes to determine which button to press in order for the application to do what you want it to do –hesitation time- is going to be recorded also. The number of “clicks” will be collected to determine how complex the system is. To determine how complex and not obvious the task procedure is, the number of errors will also be recorded.

**Brief summary of what you are going to be doing**  
First you are going to add the class that you need help on. This can be any class that you struggle with. On our current assumption, you cannot add things outside the class like riding a bicycle, or piano. After adding the class you need help on, then you will search the helpers based on that class.

To add a class, first you need to go to the list of the classes entered to modify it to make it to your preferences. Then add the class on that list. Pressing on the class title, you will be forwarded to the list of the helpers who can help with that class. From there, you can send messages to people to contact them and arrange for the time and location, but this is not required in this case of our interface comparison.
Time it takes
It will roughly take about minimum of 30 to maximum of 60 minutes.

#Attachment 2 : raw data collected

<table>
<thead>
<tr>
<th></th>
<th>Subject #1</th>
<th>Subject #1</th>
<th>Subject #2</th>
<th>Subject #2</th>
<th>Subject #3</th>
<th>Subject #3</th>
<th>Subject #3 (del)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Time</td>
<td>8.6</td>
<td>32</td>
<td>5</td>
<td>36</td>
<td>4</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Error recovery</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>23</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>#’s click</td>
<td>2</td>
<td>8</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Subject #1</th>
<th>Subject #1</th>
<th>Subject #2</th>
<th>Subject #2</th>
<th>Subject #3</th>
<th>Subject #3</th>
<th>Subject #3 (del)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Time</td>
<td>22</td>
<td>22</td>
<td>14</td>
<td>27</td>
<td>10</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>Error recovery</td>
<td>12</td>
<td>18</td>
<td>9</td>
<td>18</td>
<td>5</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>#’s click</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>9</td>
<td>3</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

#Attachment 3 : Visualization of Date collected