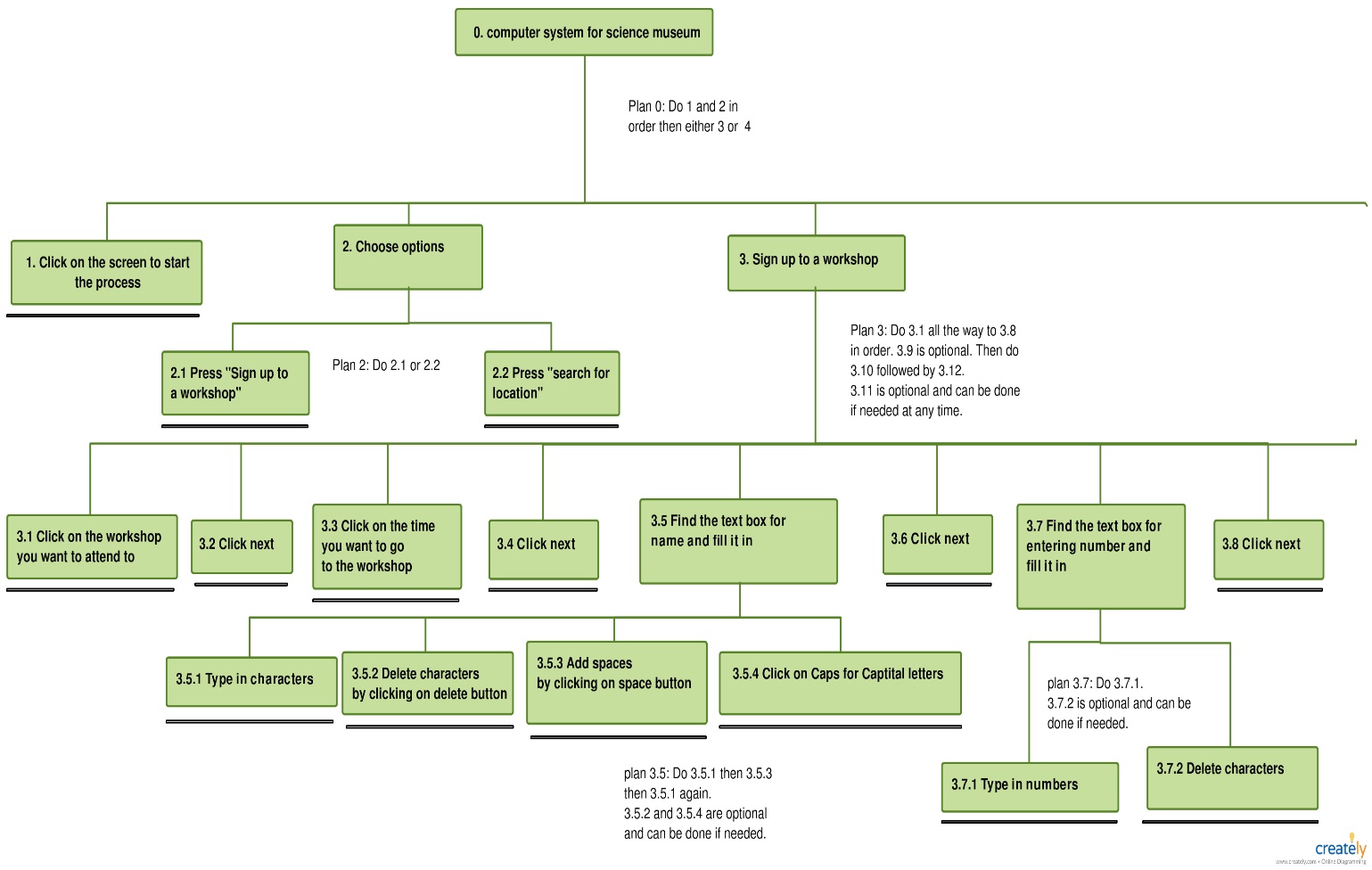
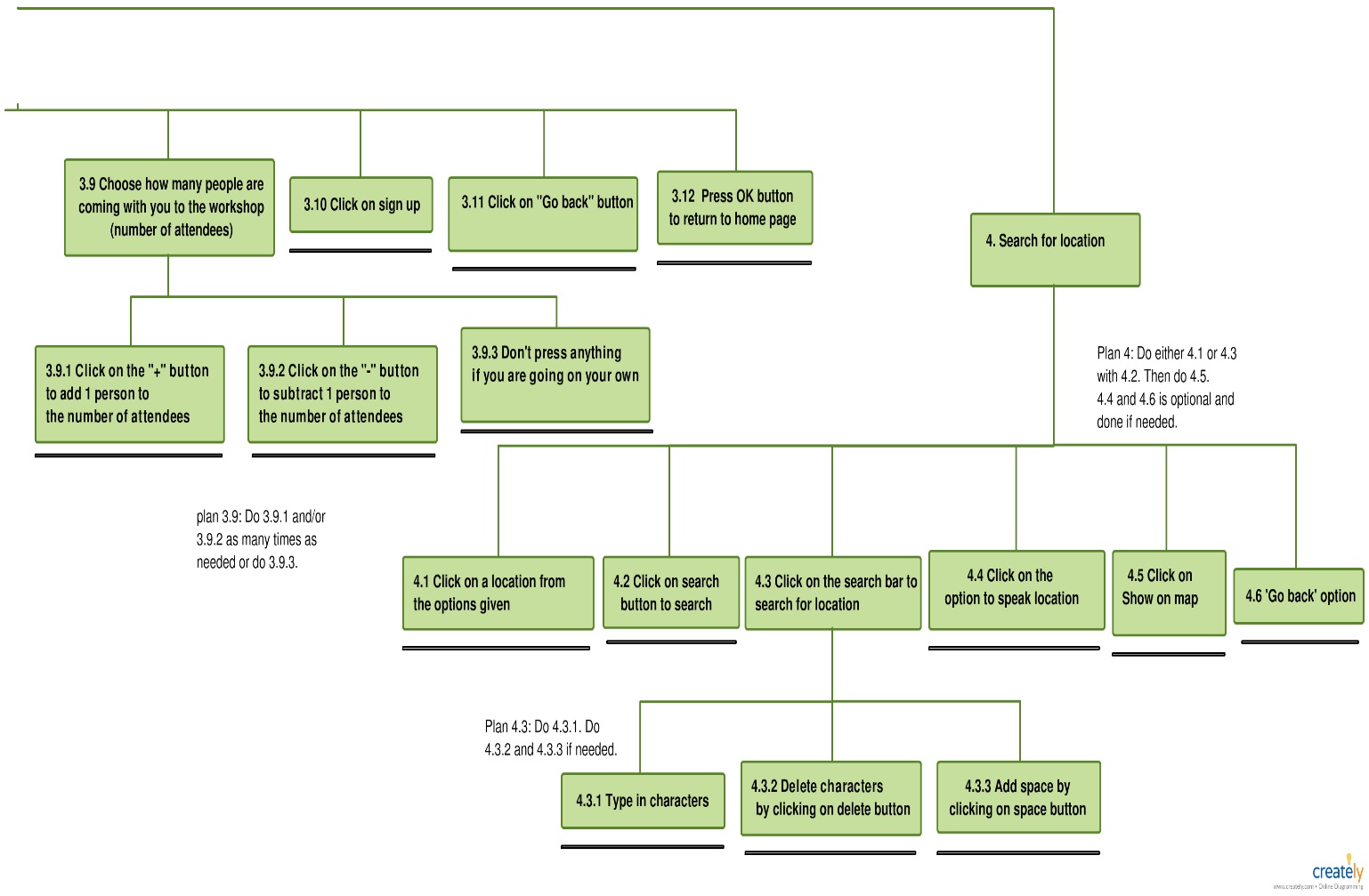
1. **Hierarchical Task Analysis diagram covering an interaction design for the above functions:**

**2. Error Analysis table detailing predicted error modes and proposed design solutions:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task** | **Error mode** | **System consequence** | **Error recovery** | **implication** |
| 1 | A7 Right operation on wrong object | The user can’t continue with the process. Button is not being clicked on. | Recovery depends on the system notifying the user. | Design: Make the button fit the entire form so it will continue if anywhere on the form is clicked on. |
| 2.1/2.2 | S2 Wrong selection made | The user proceeds in the wrong path. | Pressing the back button to return to the previous task. | Design: Make the options as clear as possible so that making an error is very slim. |
| 3.1 and 3.3 | A7 Right operation on wrong button | The user proceeds in the wrong path with the wrong information. | Pressing the back button to return to the previous task. | Design: Having a Label that lets the user know what they have clicked on before continuing. |
| 3.1 and 3.3 | S1 Selection omitted | The user continues without choosing a workshop or time. | Pressing the back button to return to the previous task and choosing a workshop and time. | Design: Not allowing the user to continue and having a label that notifies the user to choose a workshop and time before continuing. |
| 3.2, 3.4, 3.6, 3.8, 3.10, 3.12, and 4.5 | S1 Selection omitted | The process cannot continue since the button hasn’t been clicked on. Form might return to attractor screen after some time. | Recovery depends on the system notifying the user. | Design: Make the button flash or have a label telling the user to click on the button to continue. |
| 3.5.1, 3.7.1 and 4.3.1 | S2 Wrong selection made | Incorrect characters or numbers are typed when filling in the information required. | Having the ability to delete characters and numbers and re-entering other characters and numbers. | Design: Making the buttons on the qwerty keyboard and the numeric keyboard large enough to reduce or eliminate the error. |
| 3.5.2, 3.7.2 and 4.3.2 | S2 Wrong selection made | Character(s) or number(s) that are needed are deleted. | Being able to re-enter the same character(s) or number(s) again. | Design: Show in real time when a character or number is deleted. |
| 3.5.4 | S2 Wrong selection made | Causing the next character to be in capital or in lowercase when it isn’t needed. | Having the ability to delete Characters and re-entering them in the right state (capital or lowercase). Also having the ability to re-clicking on the button to return it to its preview state. | Design: Causing the Caps button to change colour to cadet blue to show the next Character is going to be capital. |
| 3.9.1, 3.9.2 and 3.9.3 | S2 Wrong selection made | Chooses the wrong number of people who are attending to the workshop. | If the number of people attending the workshop is more than intended, then click on the “- “button. If you want more people to attend with you, then click on the “+” button. | Design: Have a textbox to show how many people coming with you (start with 0). Also have a limited amount of space for how many people can attend a certain workshop which restricts the amount of people that you can bring. |
| 3.10 | S2 Wrong selection made | You sign up to a workshop but chose the wrong number of people coming with you. | Recovery depends on whether the user can see the textbox which represents the number of people coming with him. | Allow the textbox which shows how many people coming with you to change in real time as you click on “- “and “+” button. |
| 3.11 and 4.6 | S2 Wrong selection made | Goes back to the previous page which causes any information that was going on to be lost. | Re-doing that page if needed and clicking on the button which allows it to move forward (going to the page that it was on). | Design: Placing the back Button at the top left corner of the form to minimise accidental touching. Also keeping the information that was on that form even after pressing the back button and only deleting the information once the process is complete or the timer is activated. |
| 4.1 | R2 Wrong information obtained | Incorrect information is displayed to the user. | Having the ability to click on a different location and seeing the information change to the location clicked. | Design: Making sure that the name of the location is also added when giving directions to the location. |
| 4.2 | A11 Operation too early / late | No feedback in the textbox. | Depends on how long it takes for the user to click on the search bar and do a search. | Design: Make Textbox inform the user to type something before clicking on search. |
| 4.1 or 4.2 | S1 Selection omitted | User won’t get any visual feedback, so no directions given. | Recovery depends on the system notifying the user. | Design: Inform the user to either click on a location or search for it to continue after a certain amount of time. |
| 4.3 | S1 Selection omitted /  I4 Information communication unclear | User can’t identify the textbox as a search bar, so the user can’t type a location. | Depends on the level of understanding the user has and whether they can identify the textbox to be a search bar. | Design: Add a Search icon to the textbox and place the search bar at the top of the form so it is the first thing the user sees. |
| 4.3 | A13 Misplacement | The user can’t see what they are typing since the keyboard hides it. | Being able to move the keyboard form around to adjust its placement. | Design: make the keyboard appear under the search bar so it doesn’t hide it. |
| 4.3 | S2 Wrong selection made | The keyboard will hide the location buttons, so user can’t proceed. | Depends on how quick the user can get rid of the keyboard form by clicking on the “X” button on top right corner of the form. | Design: Not hiding the “X” button on the top right corner of the form so the user can click on it to get rid of the keyboard form. |
| 4.4 | A11 Operation too early / late | The user won’t hear anything since the user didn’t choose or searched for a location | Visual feedback from the textbox telling the user to search for or choose a location. | Design: allow the user to hear the visual feedback along with seeing it on the textbox. |
| 4.5 | A11 Operation too early / late | The user won’t see what he expected (a map) which could cause confusion. | Depends on how long it takes for the user to realise that they must first search or choose a location. | Design: Make the textbox inform the user that they need to search or choose a location before clicking on the button. |
| Every Form/ all steps | S2 Wrong selection made | Getting rid of/ minizine the page and being in a place where the user shouldn’t be. | Depends on if the user is familiar with the program to start the process again. | Design: get rid of the 3 buttons on the top right corner of all forms (expect keyboard form) so the user won’t be able to get rid of the form that way. |

**3a. Prototype design:**

Before making diagrams of my prototype screen design and building my prototype. I need to do some research in order to produce an appropriate design that is both easily usable and consistent.

The first thing I will be looking at is button size. From the research that Seungyon Lee and Shumin Zhai did on the performance of touch screen soft button, they discovered through several experiments that “the performance of finger operated touch screen soft buttons deteriorated when the size of the button falls below a fraction of the finger width (Zhai & Lee, 04/09/2009).” Since the “anthropomorphic average width of the index finger and the thumb for adult men are 18.2 mm and 22.9 mm respectively and women 15.5 mm and 19.1 mm respectively (Zhai & Lee, 04/09/2009)”, they concluded that as long as the button size is larger than the average width of the index finger and thumb for an adult male, the button size will be acceptable, i.e. anything larger than 23mm.

Another aspect to consider when determining the size of the button is finger contact angle. This will depend on the placement of the buttons on the screen and the angle of the device to the user. T.Nishimura, K.Doi, and H.Fujimoto concluded in their research that “designing buttons with a size of 12.5 mm or more and spacing of 0 mm enables accurate and fast operation regardless of the forefinger contact angle (Nishimura, Doi, & Fujimoto).” This means that the placement of the buttons won’t have much effect on the user using the device as long as the button size is larger than 12.5mm. This also agrees with the preview research that was done by Seungyon Lee and Shumin Zhai.

Overall, it seems that there is no correct answer when it comes to choosing the right button size. However, it seems that as long as the button size is at least equal to the size of the ninety fifth percentile male distal joint breadth as recommended by the ISO 9241-9 (Nishimura, Doi, & Fujimoto), then that would be an acceptable button size. In conclusion, I decided to make sure all my button sizes were larger than 23mm. However, I would have to keep in mind that not all button sizes will be the same since each button will be for a different purpose. Therefore, I would have to make sure to keep the sizes of the button as consistent as possible by making the buttons that have the same purpose equal in size.

The layout of the keyboard that I will design is also going to be important. There are lots of different keyboard layouts to choose from. The most popular of them being qwerty keyboard, alphabetic keyboard and azerty keyboard layouts. However, from the research done by J.Noyes, it looks like the qwerty keyboard is “dominating the computer keyboard market (Noyes, June 1998).” The research show that “over the last century, there has been a large number of keyboards developed in order to challenge this supremacy, and much empirical work and investment of time have been carried out on other layouts and designs (Noyes, June 1998).” Despite that, the qwerty keyboard is still accepted as “the de facto keyboard.” Since the qwerty is “dominating the computer keyboard market”, that means that most people that buy a keyboard choose this layout.

Some of the reasons that J.Noyes mentioned that made the qwerty the best choice is; “its early arrival in the marketplace and subsequent ubiquity”, which means it’s been in use for a long time; “the familiarity that individuals have with the QWERTY layout” which allowed individuals to prefer its use even over alphabetical keyboard layout; and its resistance to change since “it is quite likely that even with a better design, individuals would be reluctant to use it (Noyes, June 1998).”

Another research has been done on mobile keyboard layout that compared Twiddler and mini-qwerty keyboards to see which would be more effective when there is limited visual feedback. From the research that was done by James Clawson, Kent Lyons, Thad Starner, and Edward Clarkson, it seems that “both the Twiddler and mini–QWERTY keyboards offer rapid mobile text entry. However, mini–QWERTY keyboards offer the further advantage of being very fast to learn assuming the user knows how to type on a desktop QWERTY keyboard (Clawson, Lyons, & Starner).” This therefore shows that even if the keyboard was hidden or not clearly seen for whatever reason, i.e. visual impairment or sun reflecting on the device which effects the screen, it is best to use the qwerty keyboard layout to reduce error and allow for easier use of the keyboard. This is why, in my design, I decided to go with the qwerty layout for my keyboard.

When it comes to choosing my numeric/phone number keyboard layout, I had two options. One was the layout which is shown in the qwerty keyboard layout. The second was the telephone keypad layout. I decided to go with the telephone keypad layout since the only function of the numeric keyboard is to allow the user to enter their phone number. The 2 most common places we enter numbers is either on our mobile phones or on a house phone and they both have the telephone keypad layout. Therefore, I chose that layout (telephone keypad layout) since the user will most likely be more familiar with it. This will reduce the chances of an error occurring and improve efficiency when it comes to completing the task.

Now that both the button size and keyboard layout are determined, it is time to think about the font size and type that I will be using. We first need to think about the people that will be using the device. Since it will be in a museum, and museums are a place that anyone from all ages can visit, we will assume that everyone will be using this device. From young children to elderly people, people suffering from dyslexia to people with vison impairment and even blind people.

There was a research done by Michael Bernard, Chia Hui Liao, & Melissa Mills that considered the effect of font size when it came to the legibility and reading time for older Adults (from 62 years to 83). They compared two font sizes which were 14 and 12 and found that font size 14 was “more legible, promote faster reading, and was preferred to the font size of 12 (Bernard, Liao, & Mills).”

Another research by Luz Rello, Martin Pielot, and Mari-Carmen Marcos looked at a similar topic which was the Effect of Font Size and Line Spacing on Online Readability. However, they experimented with a wider range of font sizes and measured “objective and subjective readability and comprehension of Wikipedia articles (Rello, Pielot, & Marcos).” They concluded that, when using font size “up to a font size of 18 points, subjective and objective readability as well as comprehension improved continuously (Rello, Pielot, & Marcos).” However, it seemed that once we got to font size 22 points, there are no further benefits to increasing the font size.

A third research, which was also done by the same three authors along with Roberto Carlini, agreed with the previous research. This one was focused on determining a Dyslexic-friendly font size. They came to the same conclusion that “up to a font size of 18, subjective and objective readability and comprehension improved (Rello, Pielot, Marcos, & Carlini).” However, anything beyond that found “no further increases for the objective measures (Rello, Pielot, Marcos, & Carlini).”

Therefore, when I build my prototype, I will be using font size 18 points for everything. This will allow for consistency throughout the operation and will be an optimal size where older people and people suffering from dyslexia can read without having too much problems.

Another factor to consider is font type since “font types have an impact on readability of people with dyslexia (Rello & Baeza-Yates).” Since everyone from the public will be using the device, we should also assume that people with reading difficulties will also want to use it. “Worldwide, around 15–20% of the population has a language-based learning disability and likely 70–80% of these have dyslexia (Rello & Baeza-Yates).” “It is also estimated that up to 1 in every 10 to 20 people in the UK has some degree of dyslexia (Dyslexia, n.d.).” Since there is a possibility that someone with reading difficulty will be using the device, it is very important to make sure their journey through the interface is as smooth and easy as possible by choosing the appropriate font type.

A very large study, which was done by Luz Rello and Richard Baeza-Yates mentions that “most of the font recommendations come from associations for people with dyslexia, and they agree in using sans-serif fonts (Rello & Baeza-Yates).” However, they have done some experiments and gathered large amount of data to see if this was correct. They also tested 12 different fonts on people with and without dyslexia to measure objective readability and preferences. They concluded that even though “participants with dyslexia had significantly lower preferences ratings than the participants without dyslexia”, the correlations between groups were significant for all the measures (Rello & Baeza-Yates).”

When comparing the results of people with and without dyslexia, it turns out that the sans serif fonts, Helvetica, Arial, Courier and CMU, led to a significantly “better objective readability”. It allowed for a “shorter reading time and shorter fixation duration (Rello & Baeza-Yates).” For people without dyslexia, “Arial, Courier, CMU and Verdana were the best in terms of reading performance and subjective preference (Rello & Baeza-Yates).” They also concluded that it was better to use non-italic font (roman font) over italic font. This is because it “led to better reading performance for people with dyslexia (shorter fixations) and without dyslexia (shorter reading times) (Rello & Baeza-Yates).” Both groups also significantly preferred fonts in roman than fonts in italic. This shows that people with dyslexia and possibly without dyslexia have a hard time reading letters that are cursive when compared to printing writing (block letters).

This agrees with a similar research done by the same authors where they looked at dyslexia friendly font types. They also found that “when the font type was sans serif and Roman style, it led to a shorter fixation duration (Rello & Baeza-Yates).” They also experimented with other font types and found that “good fonts for people with dyslexia are Helvetica, Courier, Arial, Verdana and CMU, taking into consideration both, reading performance and subjective preferences (Rello & Baeza-Yates).” Overall, I decided that I will be using the Microsoft Sans Serif which visual studio provides since it is shown that people with and without dyslexia prefer it and is better when it comes to reading time and fixation duration.

Colour blindness is another factor that I will need to consider. However, there are different types of colour blindness and some are rarer than others. It is estimates that the most common form of total colour blindness, also known as, “rod monochromacy, is thought to only occur in around 0.003% of people or less (Total Color Blindness, n.d.).” Therefore, it is very rare that someone with total colour blindness will be using our device. Even though that is the case, I will still need to add them to my planning and think about this when choosing my colours.

A more common type of colour blindness is a type called ‘red-green’ colour vison deficiency. This type of colour blindness “affects around 1 in 12 men and 1 in 200 women (Colour vision deficiency (colour blindness)).” People that are suffering from this vison deficiency will “find it hard to tell the difference between shades of reds, oranges, yellows, browns, and greens. They will also confuse reds with black (Colour vision deficiency (colour blindness)).” There is another type of colour blindness, which is rare, called ‘blue-yellow’ colour vison deficiency which would cause people to have problems with shades of blues, greens, and yellows.

Due to these findings, it seems that as long as I stay away from or minimise the use of colours such as red, orange, yellow, brown and green, I am free to choose any other colour I want without worrying about how it will affect people with colour vison deficiency (colour blindness). Overall, I decided to mainly stick with simple colours such as different shades of white, black, and grey since these colours can be seen by everyone including people with complete colour blindness. When it comes to colour, I will also have to keep an eye on colour contrast. I will be using a formula in Arthur & Passini’s book Wayfinding which “describes a reliable calculating method to calculate the contrast difference between two colours. Their formula is based on the light reflectancy (LR) readings in percentages for each of the two colours involved (Color contrast, 2016).”

Another important thing to consider before building my programme is if it agrees with ISO 9241. It talks about ergonomic requirements for office work with visual display terminal (VDTs). Not exactly what my programmes bout but similar principles will apply. I need to make sure that my programme is follows the general interface guidelines that the ISO 9241 has such as making sure that the only thing that the user sees is the information that is related to the task the user is performing at that moment. I need to also make sure that it agrees with the user control guidelines such as allowing the programme to undo an action that was done by the user, either deleting characters or going to a preview slide. Another important guideline to follow is Error and Help Messaging Guidelines. This is important since we will need to have useful error messages that inform the user of what they should do in our programme. The Menu Guideline informs us that a good font should be used which is readable even in a distance. It also tells us to use San Serif fonts which agrees with the research I have done about font types. (Griffiths, 2017)

References

# References

Bernard, M., Liao, C. H., & Mills, M. (March 31 - April 05, 2001). The effects of font type and size on the legibility and reading time of online text by older adults. *Conference on Human Factors in Computing Systems (CHI)* (pp. 175-176 ). Seattle, Washington: AMC Digital Library.

Clawson, J., Lyons, K., & Starner, T. (18-21 Oct. 2005). The impacts of limited visual feedback on mobile text entry for the Twiddler and mini-QWERTY keyboards. *Ninth IEEE International Symposium on Wearable Computers (ISWC'05)* (pp. 170 - 177). Osaka, Japan: IEEE Xplore.

*Color contrast*. (2016). Retrieved from designworkplan: http://designworkplan.com/design/signage-and-color-contrast.htm

*Colour vision deficiency (colour blindness)*. (2016, 04 25). Retrieved from NHS: https://www.nhs.uk/conditions/colour-vision-deficiency/#types-and-symptoms-of-colour-vision-deficiency

*Dyslexia*. (n.d.). Retrieved from NHS: https://www.nhs.uk/conditions/dyslexia/

Griffiths, L. (2017). *Semester 1 Lecture HCI part notes and resources*. Retrieved from Salford Blackboard: https://blackboard.salford.ac.uk/webapps/blackboard/content/listContent.jsp?course\_id=\_88661\_1&content\_id=\_2590578\_1

Nishimura, T., Doi, K., & Fujimoto, H. (25-29 Aug. 2015). Evaluation of touch-sensitive screen tablet terminal button size and spacing accounting for effect of fingertip contact angle. *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 1148 - 1151). Milan, Italy: IEEE Xplore.

Noyes, J. (June 1998). QWERTY-the immortal keyboard. *Computing & Control Engineering Journal* , 5-6.

Rello, L., & Baeza-Yates, R. (May 2016). The Effect of Font Type on Screen Readability by People with Dyslexia. *ACM Transactions on Accessible Computing (TACCESS)*, Article No. 15 Volume 8 Issue 4.

Rello, L., & Baeza-Yates, R. (October 21 - 23, 2013). Good fonts for dyslexia. *ASSETS '13 Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility* (p. Article No. 14). Bellevue, Washington: AMC Digital Library.

Rello, L., Pielot, M., & Marcos, M.-C. (May 07 - 12, 2016 ). Make It Big!: The Effect of Font Size and Line Spacing on Online Readability. *CHI '16 Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 3637-3648). San Jose, California, USA: AMC Digital Library.

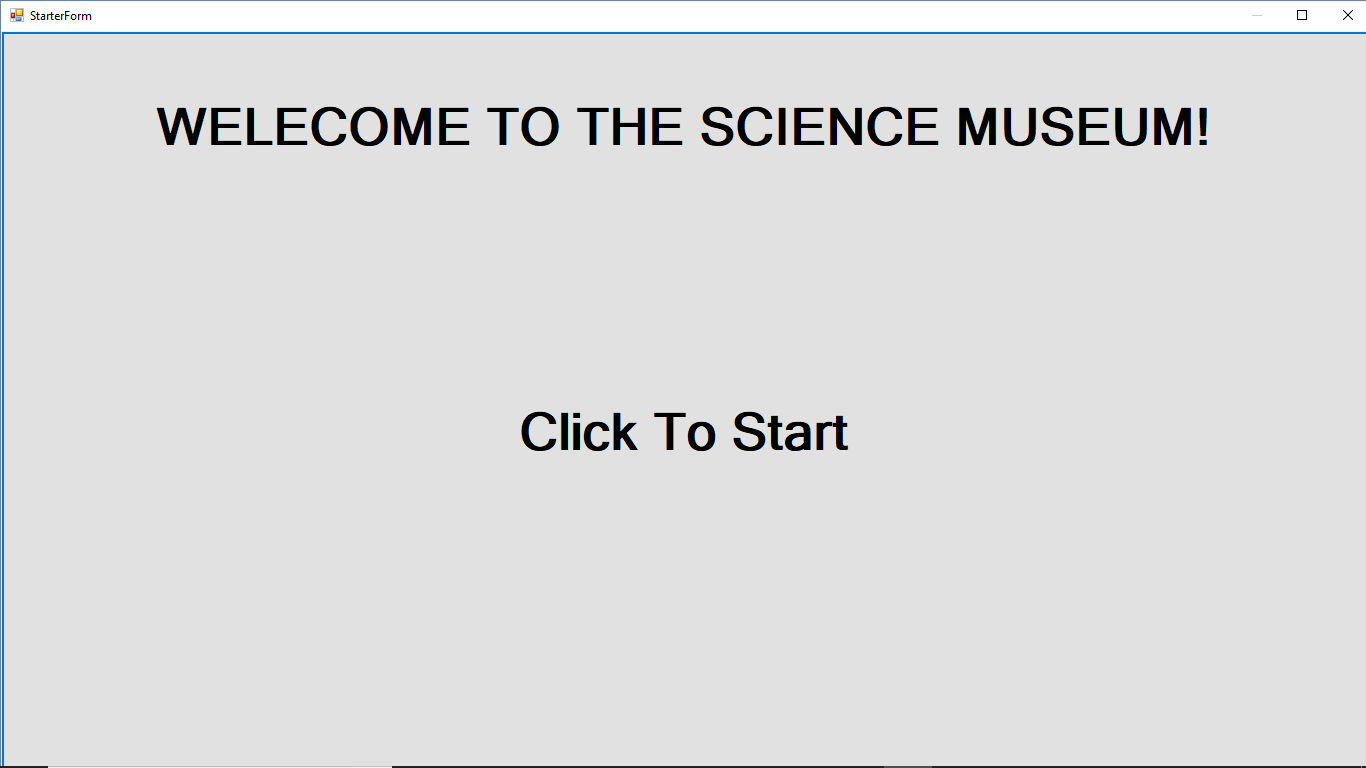
Rello, L., Pielot, M., Marcos, M.-C., & Carlini, R. (May 13 - 15, 2013 ). Size matters (spacing not): 18 points for a dyslexic-friendly Wikipedia. *W4A '13 Proceedings of the 10th International Cross-Disciplinary Conference on Web Accessibility* (p. Article No. 17). Rio de Janeiro, Brazil: AMC Digital Library.

*Total Color Blindness*. (n.d.). Retrieved from Color Blindness: http://www.colour-blindness.com/variations/total/

Zhai, S., & Lee, S. (04/09/2009). The Performance of Touch Screen Soft Buttons. *SIGCHI Conference on Human Factors in Computing Systems* (pp. 309-318). Boston, MA, USA: AMC Digital Library.

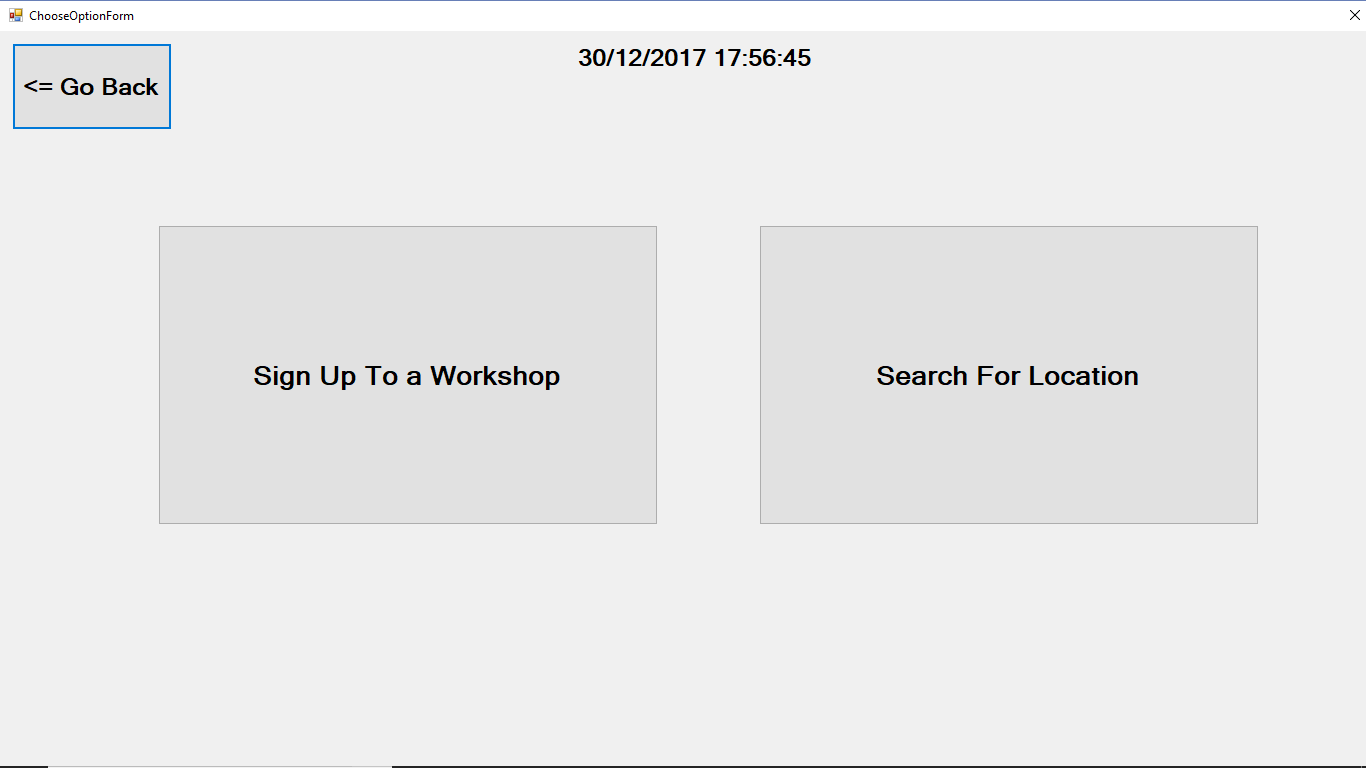
**3b. Diagrams of your prototype screen designs including any features and messages taken from**

**your error table and other design research:**



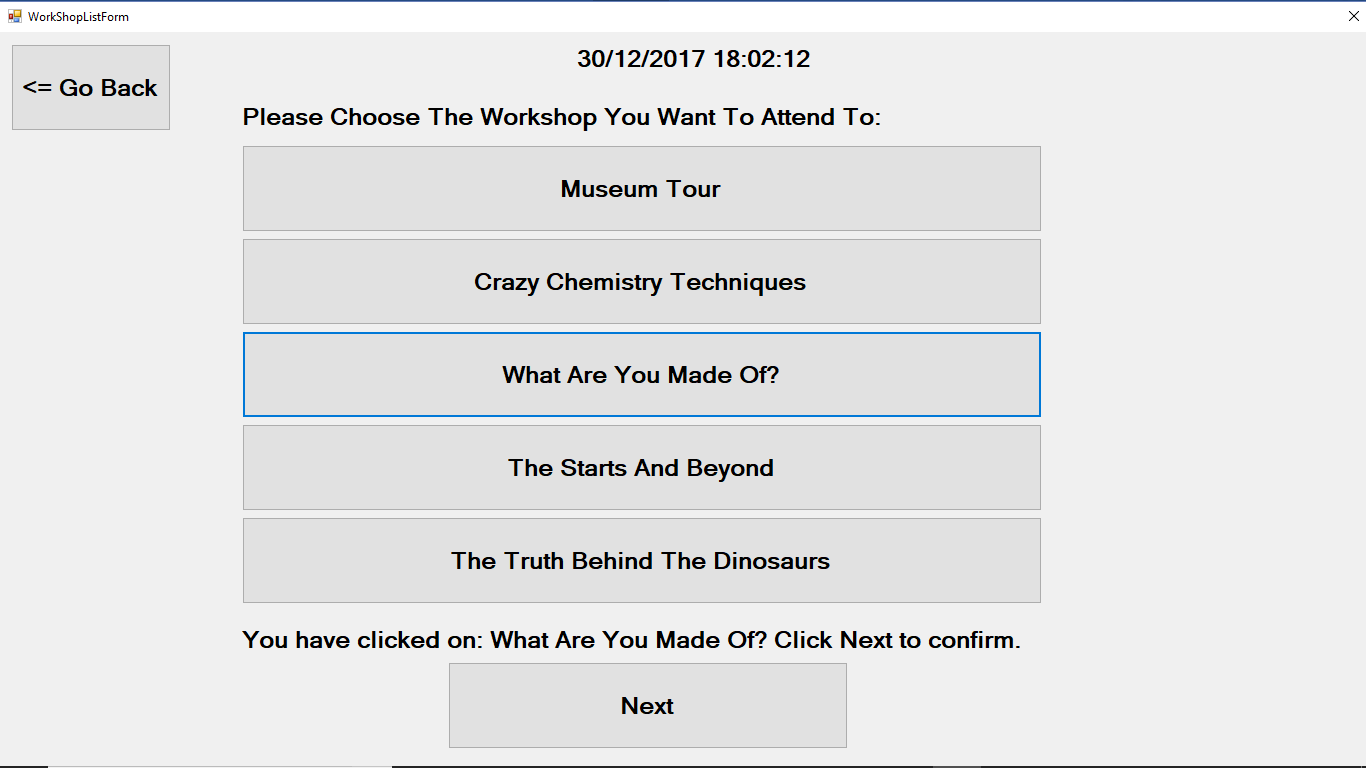
(Step 1)

This is the attractor screen. The button is bound to the entire screen of the Form. Therefore, regardless of where the user presses, it will take them to the next page.



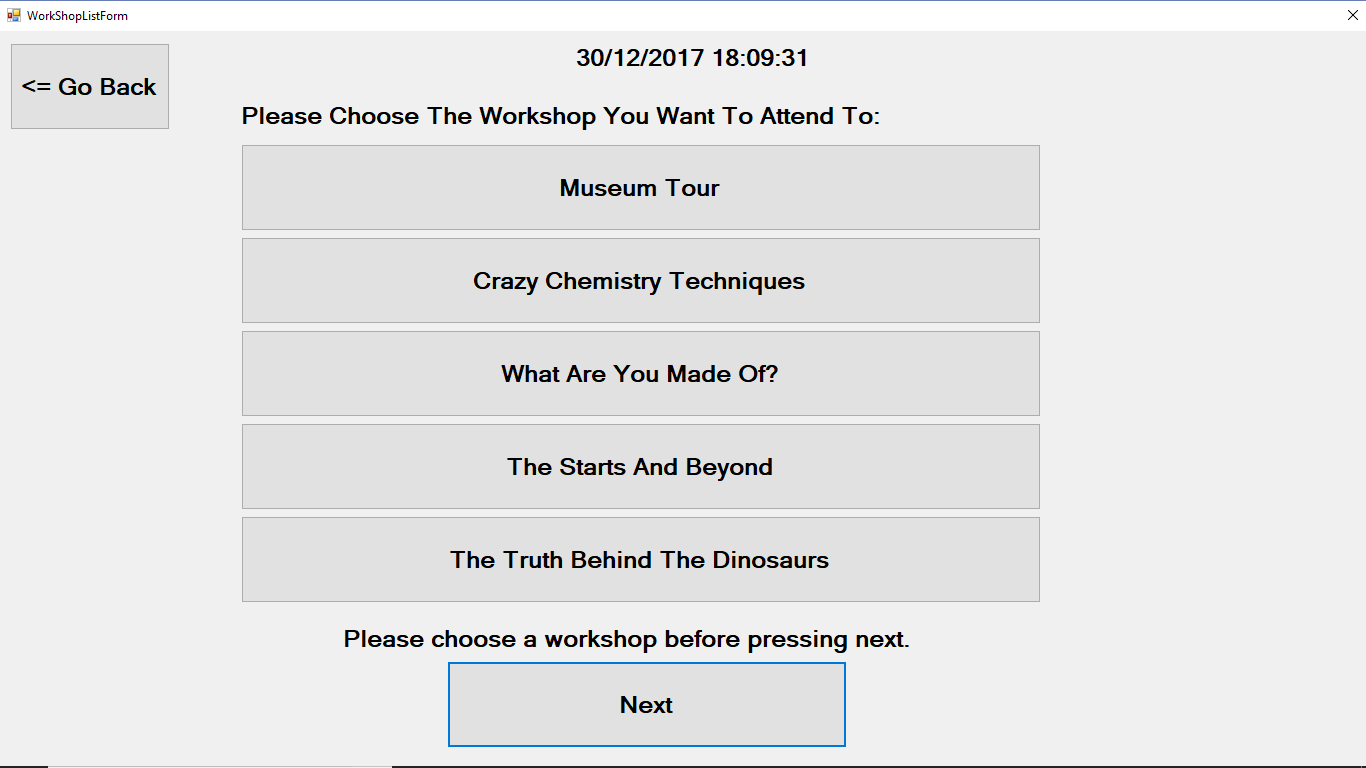
(Step 2.1, 2.2 and 3.11)

This page shows the timer and date at the top of the form which I think is useful. It also shows 3 buttons. 2 buttons, that have similar function, are the same size and are placed in the middle of the screen to attract the user’s attention. Another button (back button) is smaller in size and is placed on the top left corner of the screen for all the forms to reduce the change of accidently clicking on it. However, all 3 buttons are acceptable as shown by my research that a good button size is anything larger than 23mm.



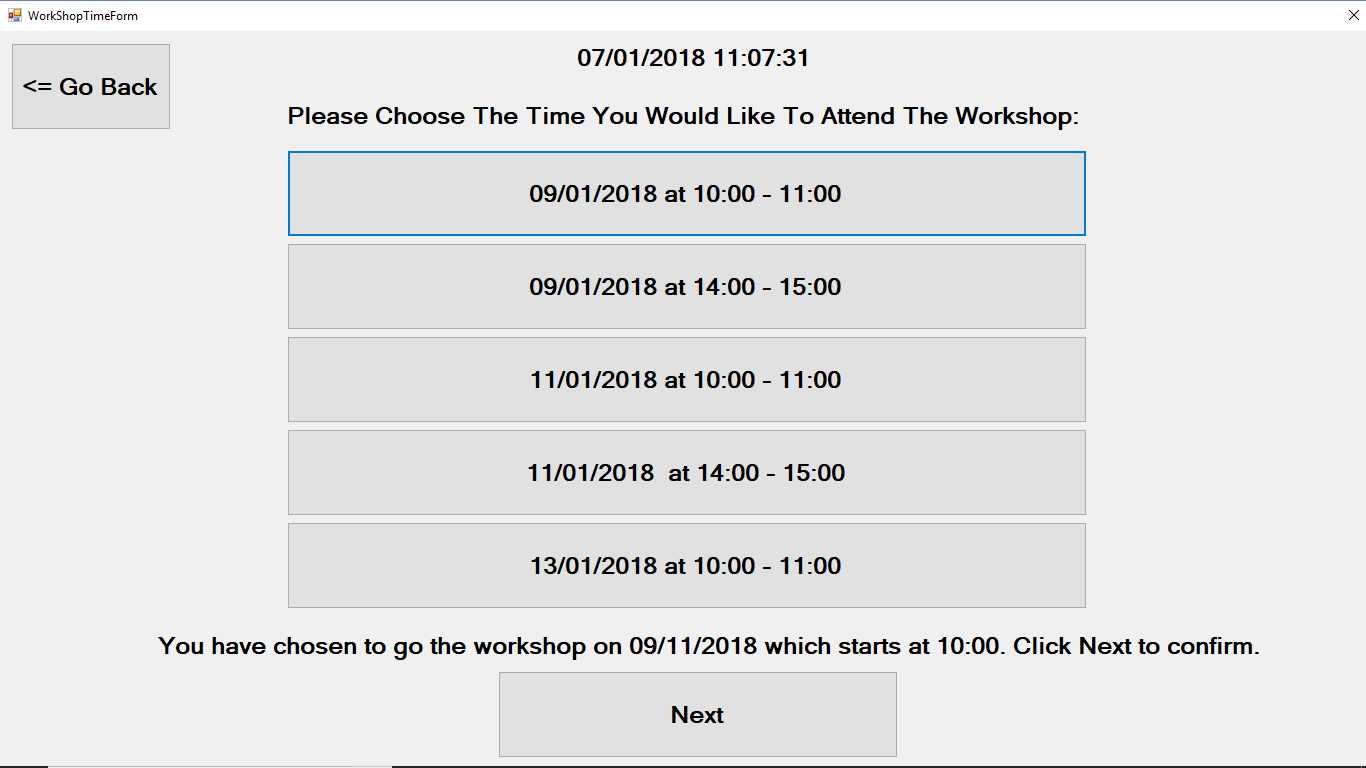
(Step 3.1 and 3.2)

Once a button (session) is clicked on, a label between the next button and sessions button shows what the user has clicked on. The next button is also placed at end of the page to say that this is the last thing that is done, i.e. pick a workshop first.

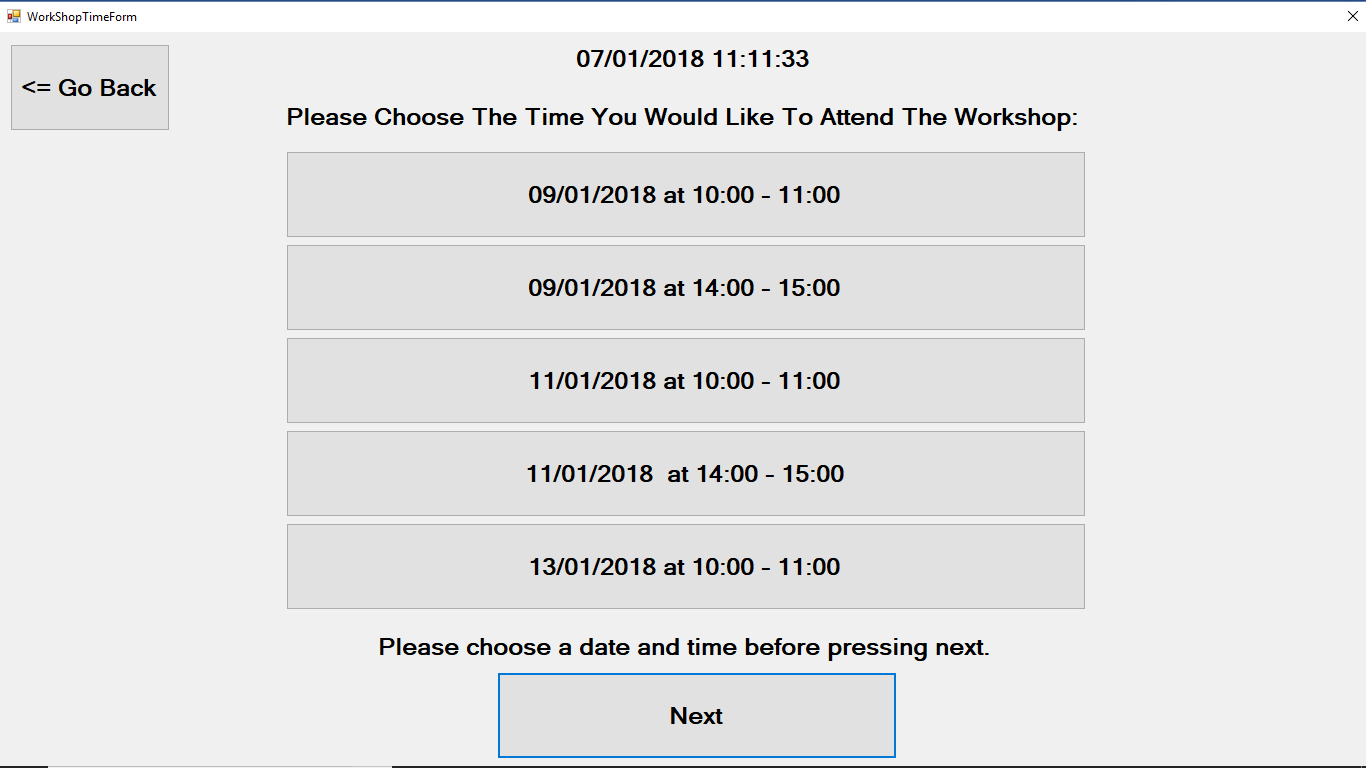


(Step 3.1 and 3.2)

If the user doesn’t choose a session (click on a session button), then a label between the next button and sessions button will appear telling the user to choose a session before clicking next.

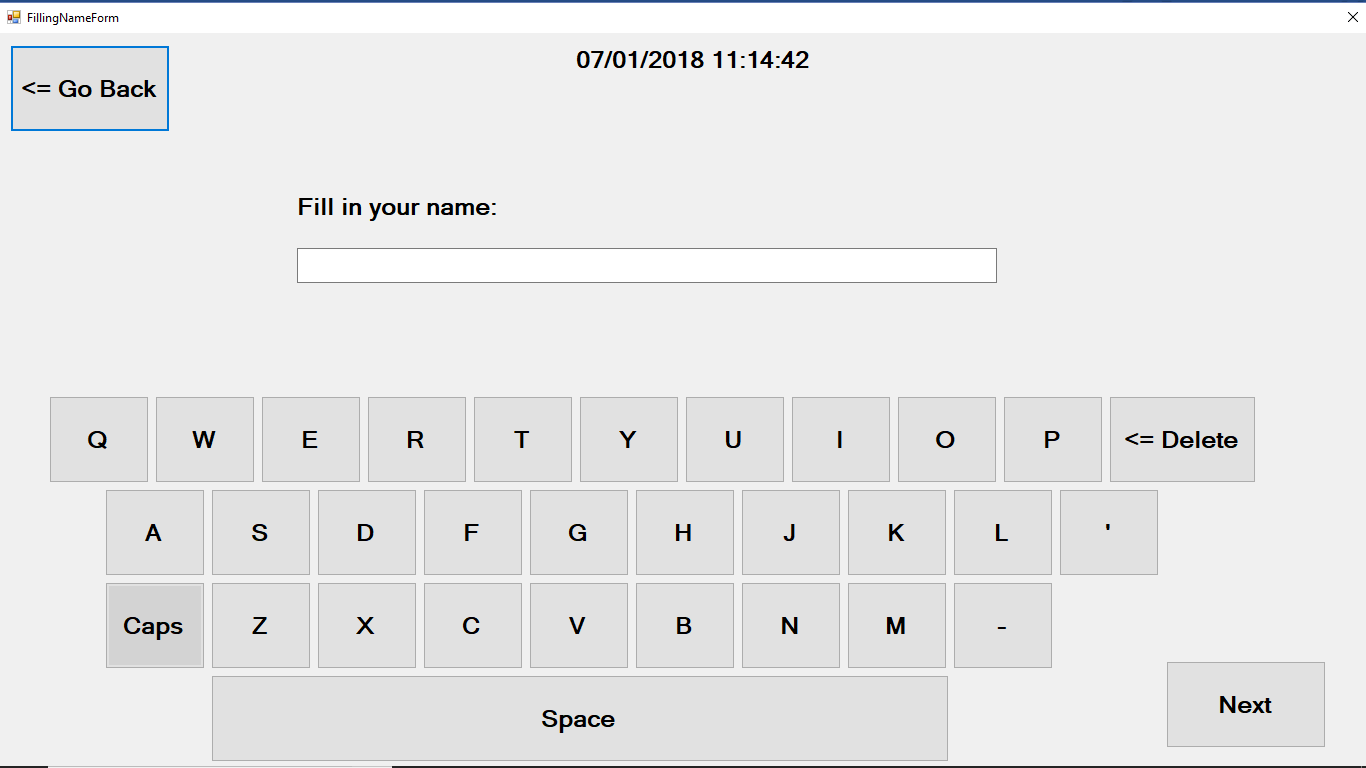


(3.3 and 3.4)

Once a button (date/ time) is clicked on, a label between the next button and times button shows what the user has clicked on.

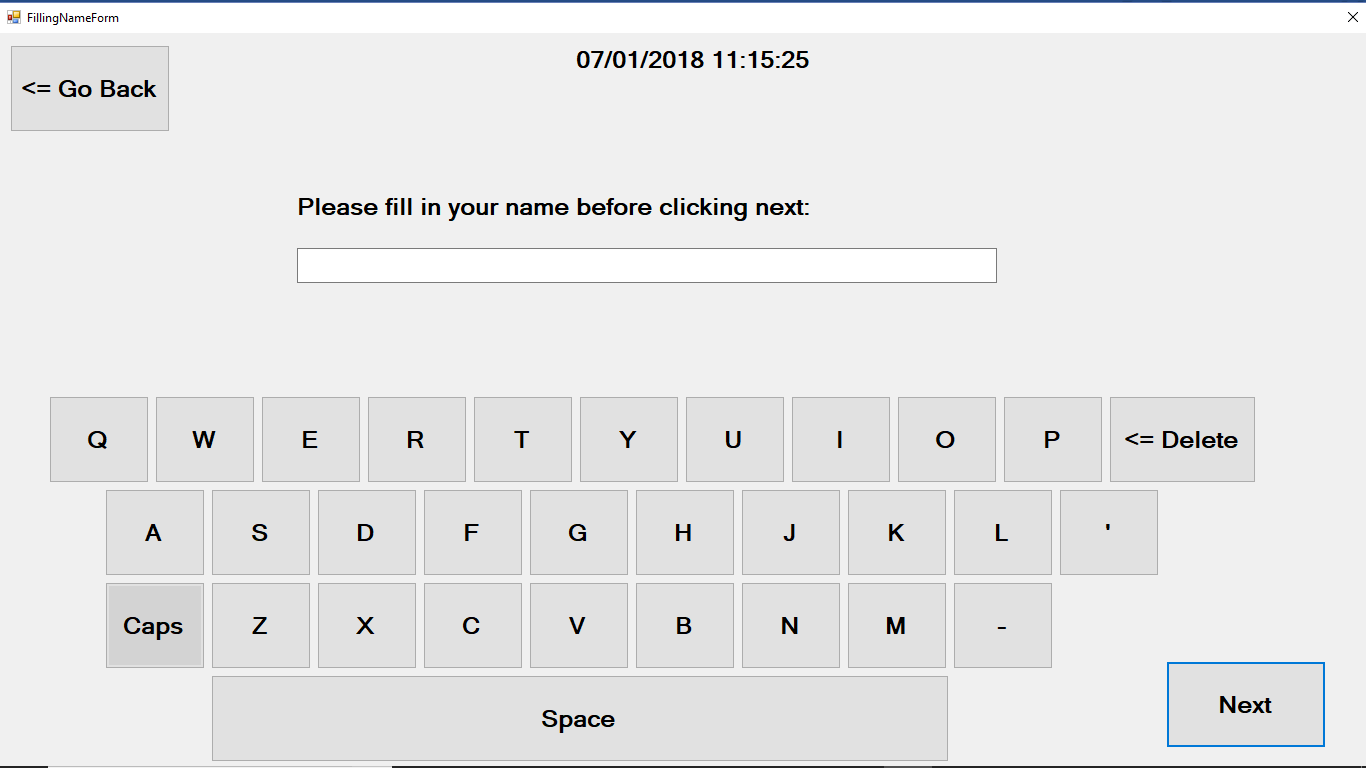
(3.3 and 3.4)

If the user doesn’t choose a session (click on a date with time button), then a label between the next button and sessions button will appear telling the user to choose a time before clicking next.

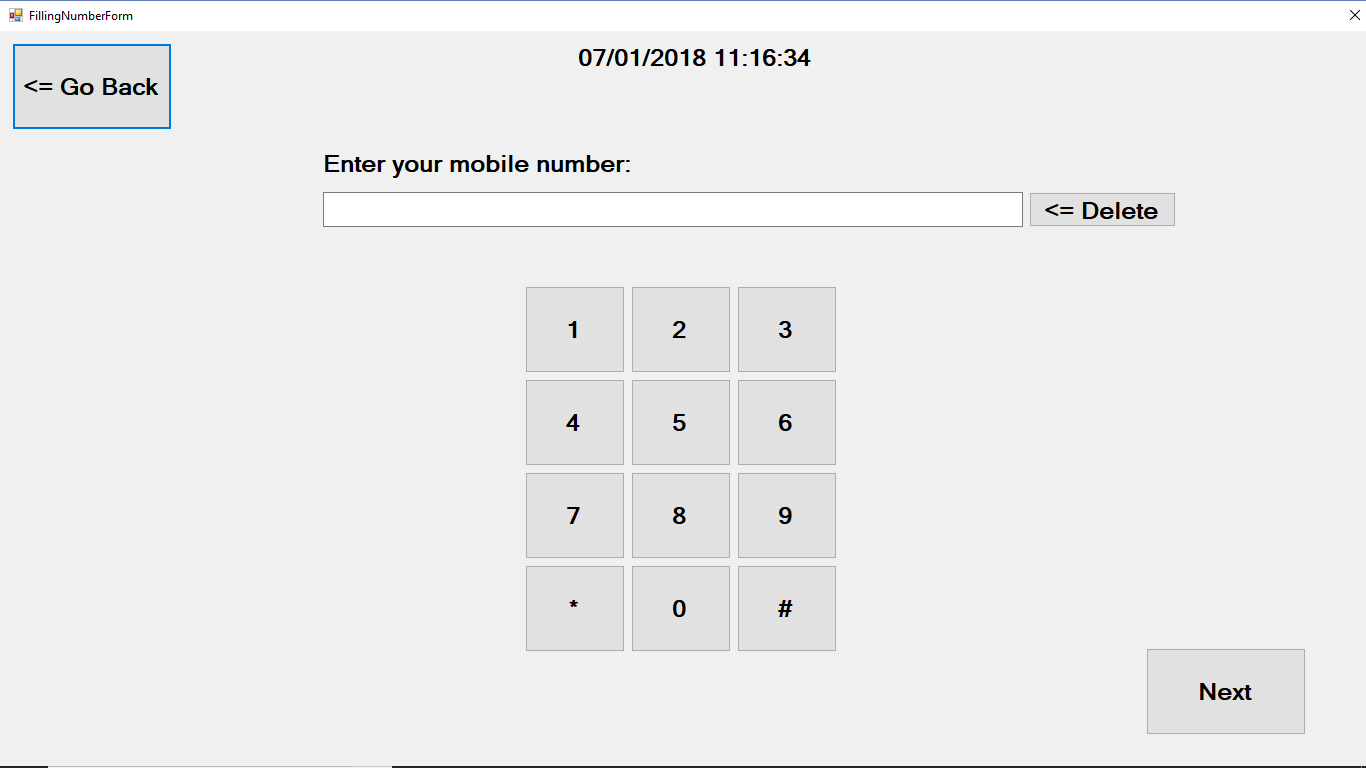


(Step 3.5, 3.5.1, 3.52, 3.53, 3.5.4)

There are several buttons at the bottom of the screen to represent a keyboard with only the needed characters, a delete button, and caps lock button. Once a keyboard button is clicked, it is shown in the name textbox shown above it. Clicking on the caps button will allow the next word to be a capital letter. It will also change colour to represent that the caps lock is active. Clicking on the delete button will remove one character (the most recent one added).

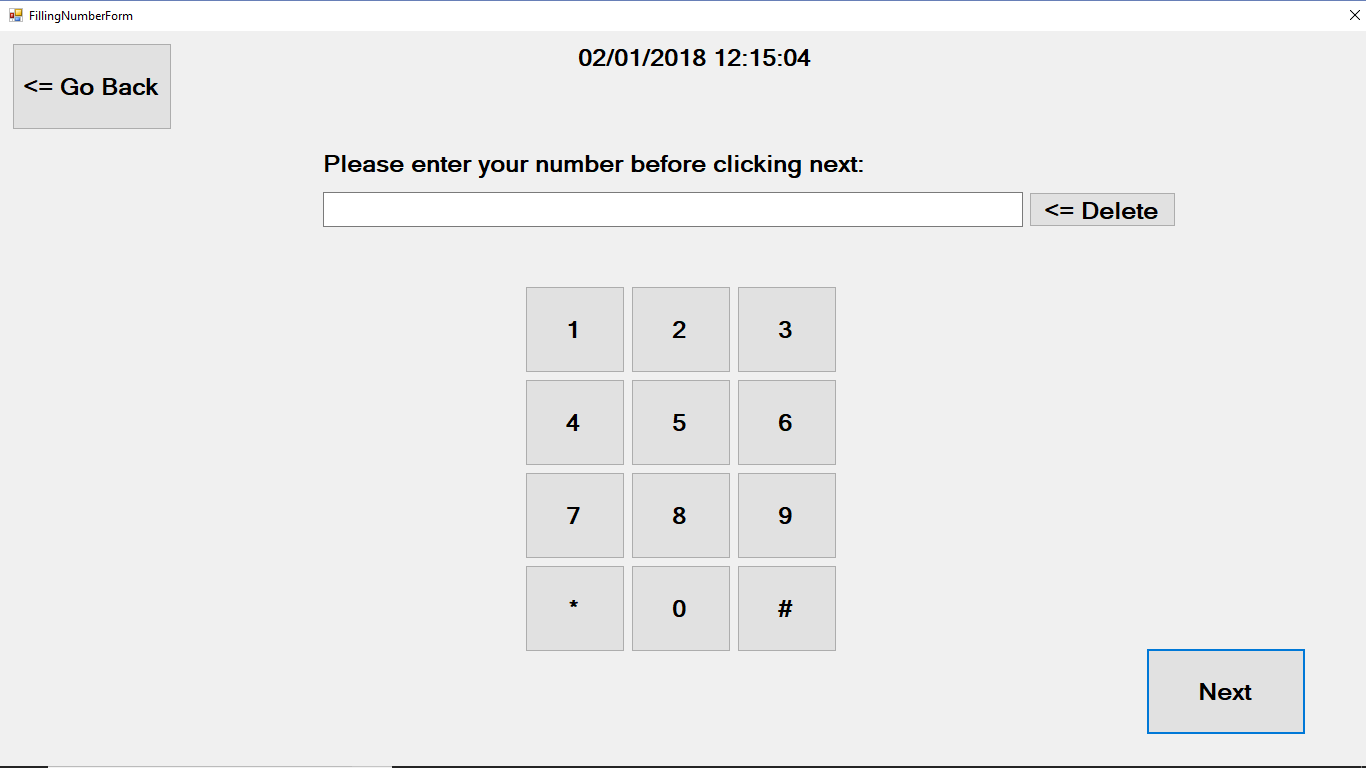


(3.6)

If the next button is clicked without typing in a name. the label above the textbox will change to let the user know to fill in their name before carrying on with the process. 

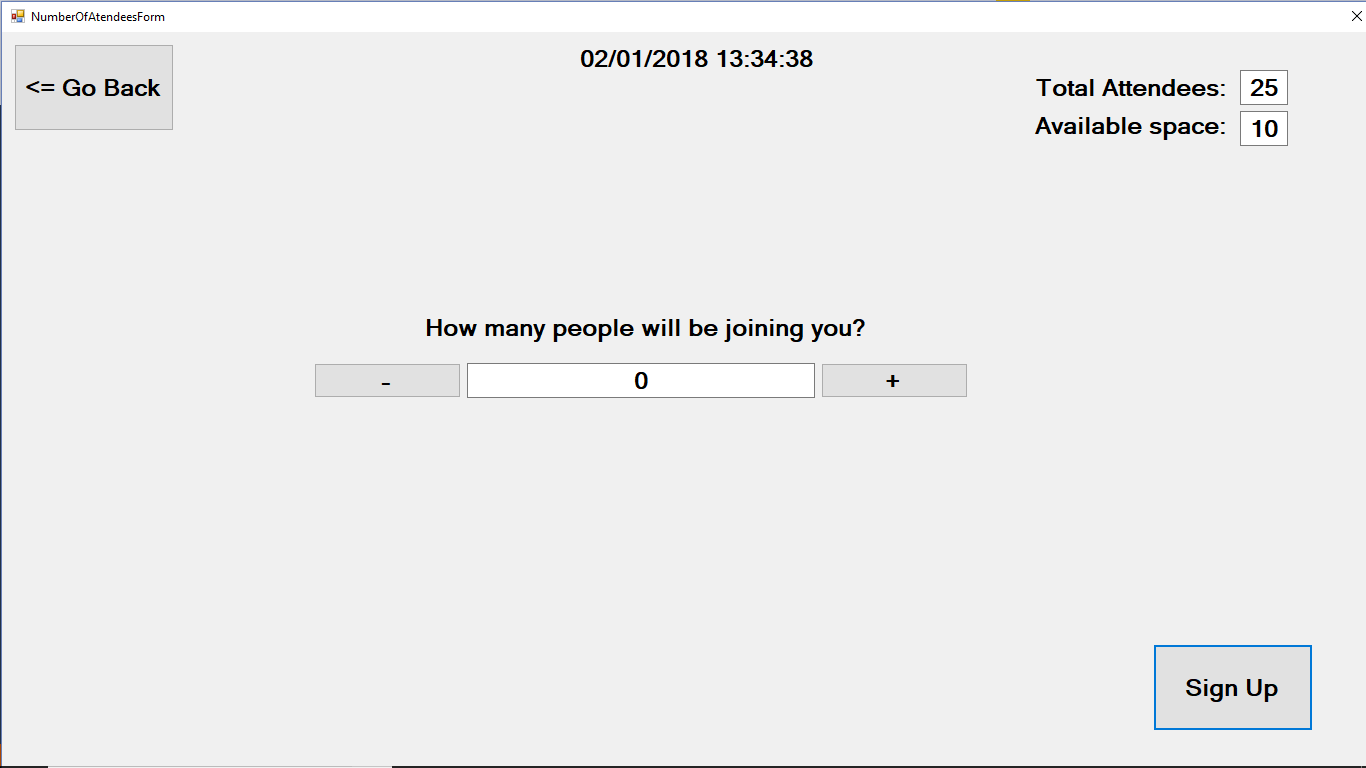
(3.7, 3.7.1 and 3.7.2)

There are several buttons of the same size which make a numeric keyboard. When number button is clicked, it is shown in the number textbox above it. Clicking on the delete button will remove one number (the most recent one added).



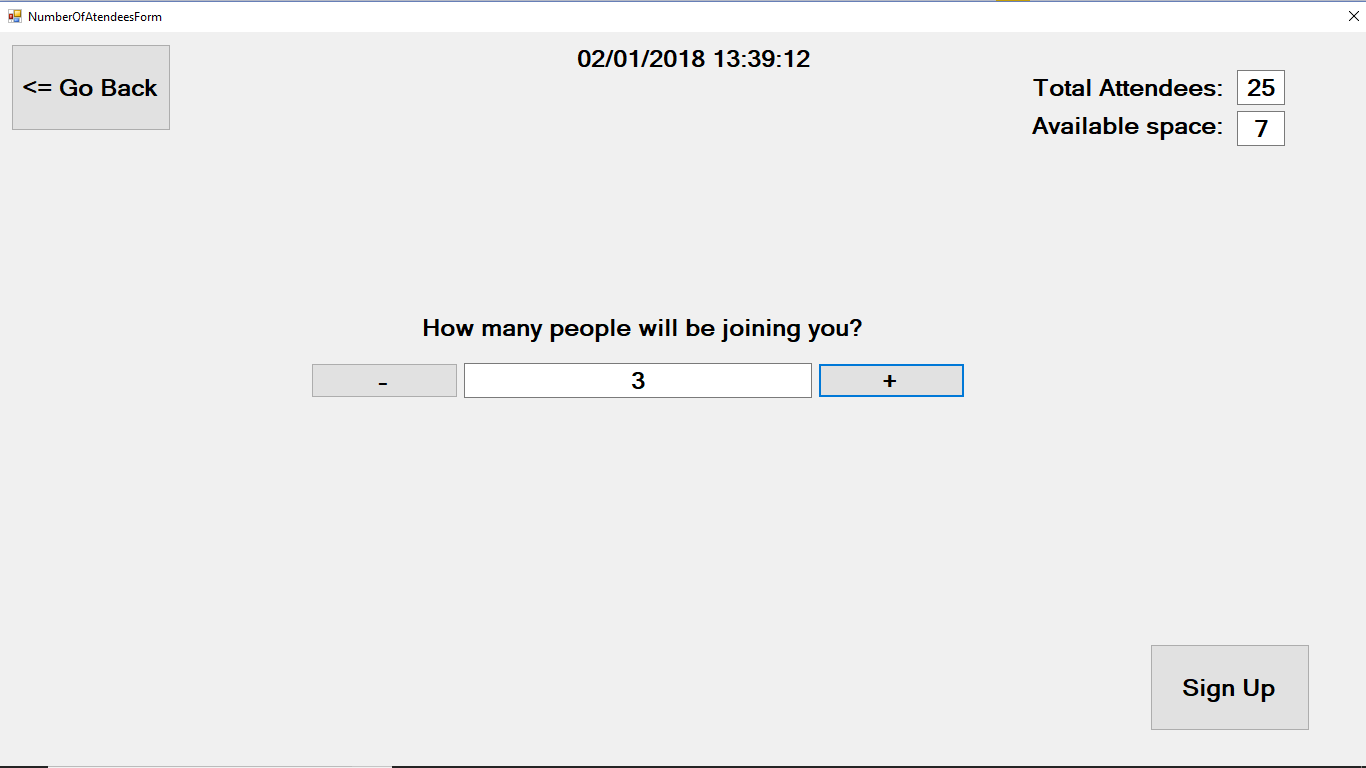
(3.8)

If the next button is clicked without entering a number. The label above the textbox will change to let the user know to fill in their number before carrying on with the process.



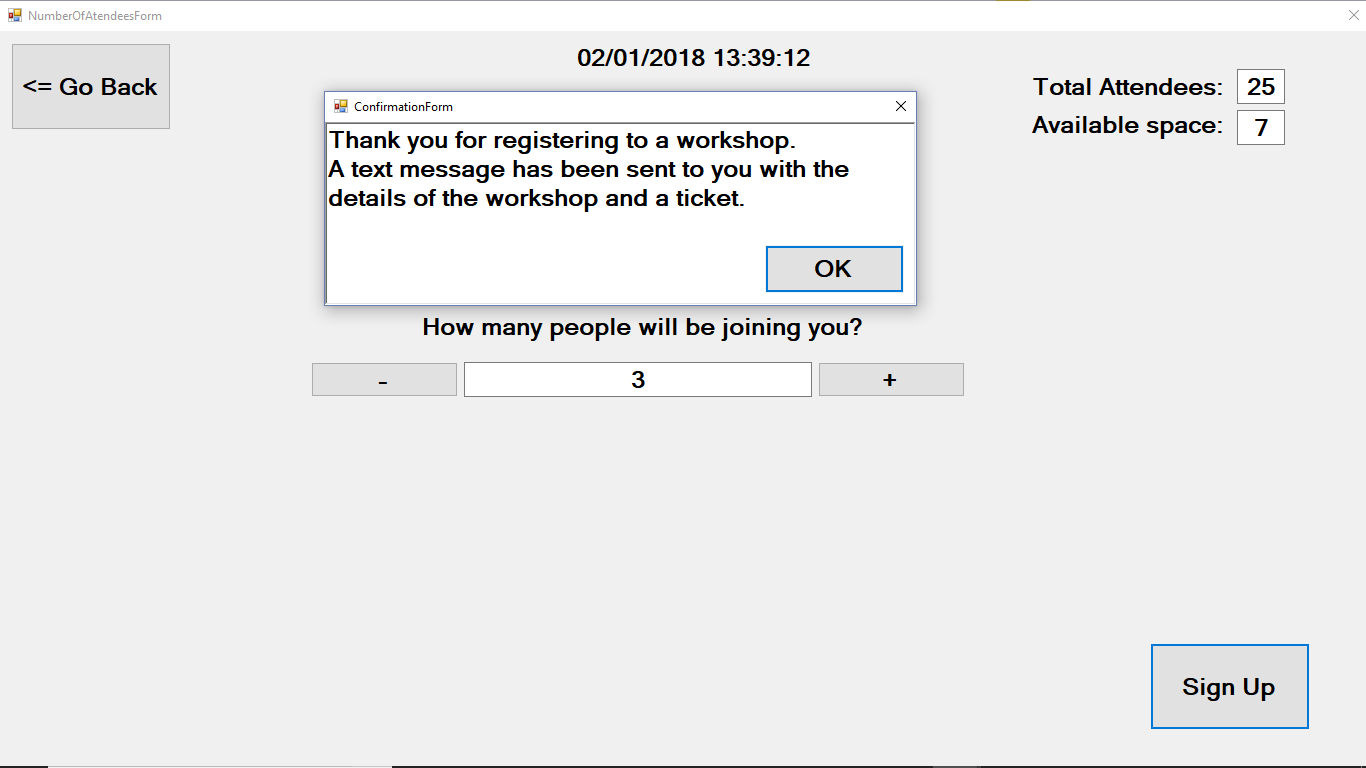
(3.9 and 3.9.3)

At the top right corner, there are 2 pairs of labels and textboxes. The top one shows the total number of people attending the workshop session while the bottom one shows the amount of spaces left. There is also a text box which is in between 2 buttons that represents that amount of people coming with you to the workshop session. One button allows the user to add a person while the other button allows the user to reduce the amount of people coming. If the textbox shows 0, then that number can’t be reduced. This form shows the user going on his own since there is 0 people going with him to the workshop session.



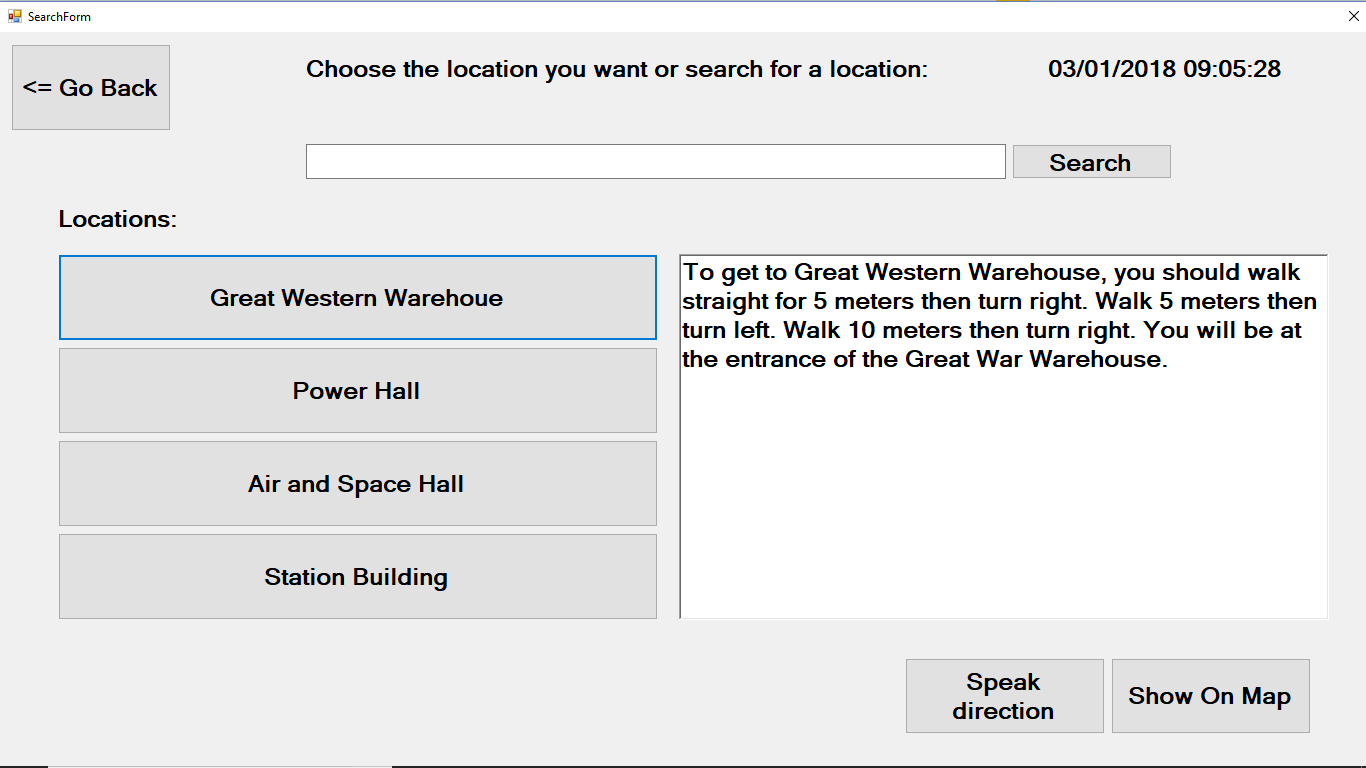
(3.9, 3.9.1 and 3.9.2)

This screen shows the user letting the system know that he wants to bring 3 people with him as seen in the textbox in the middle of the screen. This was done by clicking on the “+” button three times. The textbox on the top right corner also changed and shows that there are now 7 spaces left since you are about to fill in 3 more spaces.



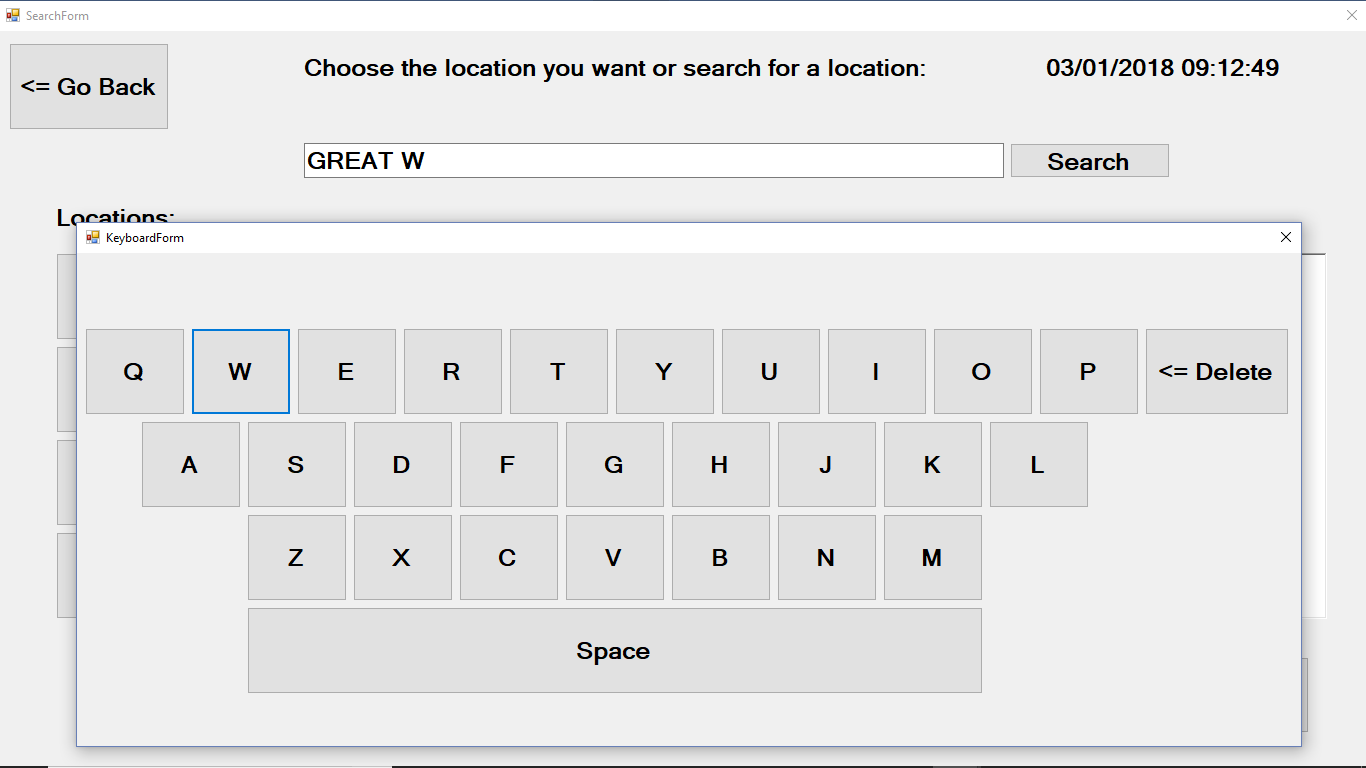
(3.10 and 3.12)

This screen shows what happens when the user presses on the sign-up button. A “message box” will appear informing the user that their registration has been successful. They will need to click on the ok button to end their journey of signing up to a workshop which will return the system to the Attractor screen. A timer is also set so that if the ok button is not clicked after 5 minutes, the screen will automatically return to the attractor screen. This timer is also set up in every other screen. This is useful because even if someone walks away half way through the process, after 5 minutes, it will return to the attractor screen to prepare it for the next person.



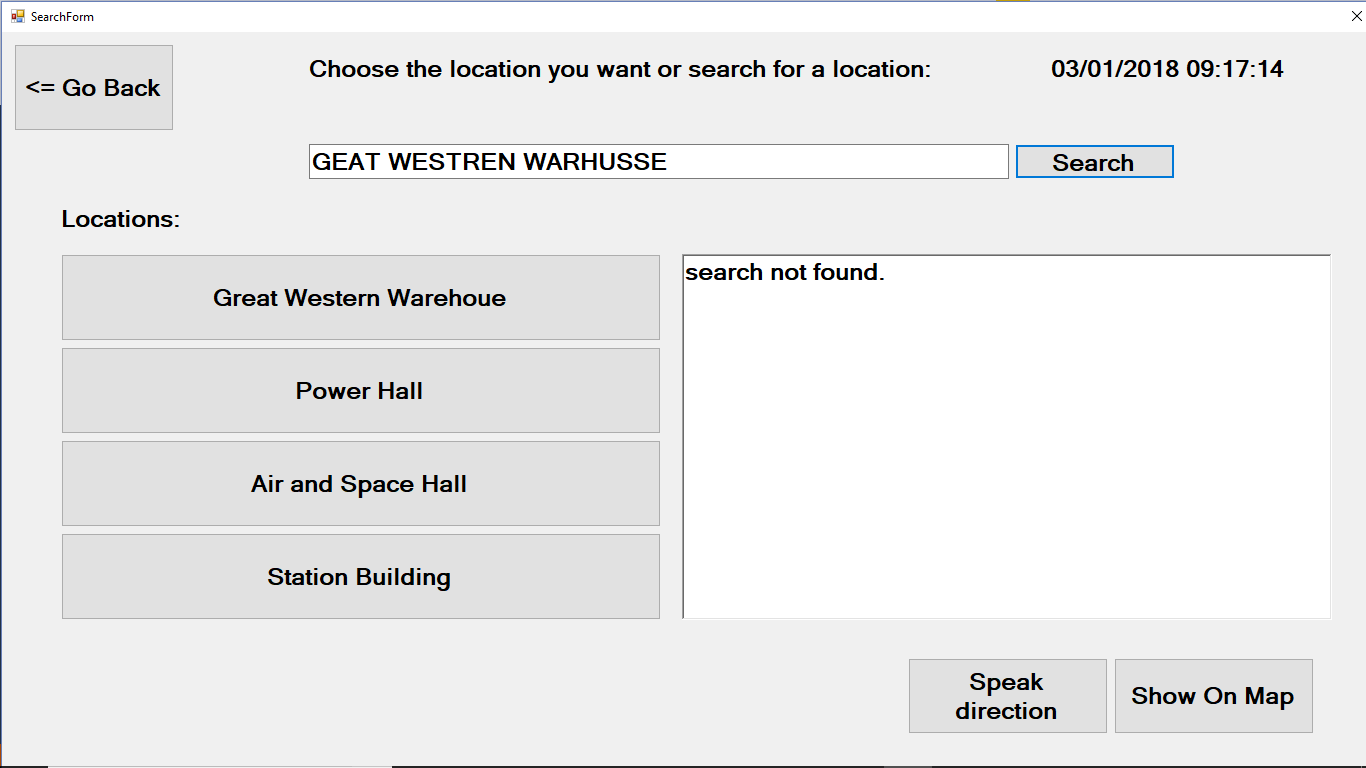
(4.1 and 4.6)

The back button is placed on the top left corner to reduce the change of accidently clicking on it. There are 4 buttons on the left side of the screen which perform a similar task. They are responsible for showing information about a certain location. The location is shown on the textbox on the right. There is also a search bar on the top with a search button. There are also 2 other buttons at the bottom of the screen which allows for speech and a visual map of the place. This screen shows the user clicking on one of the given options and the textbox giving the user direction to the place.



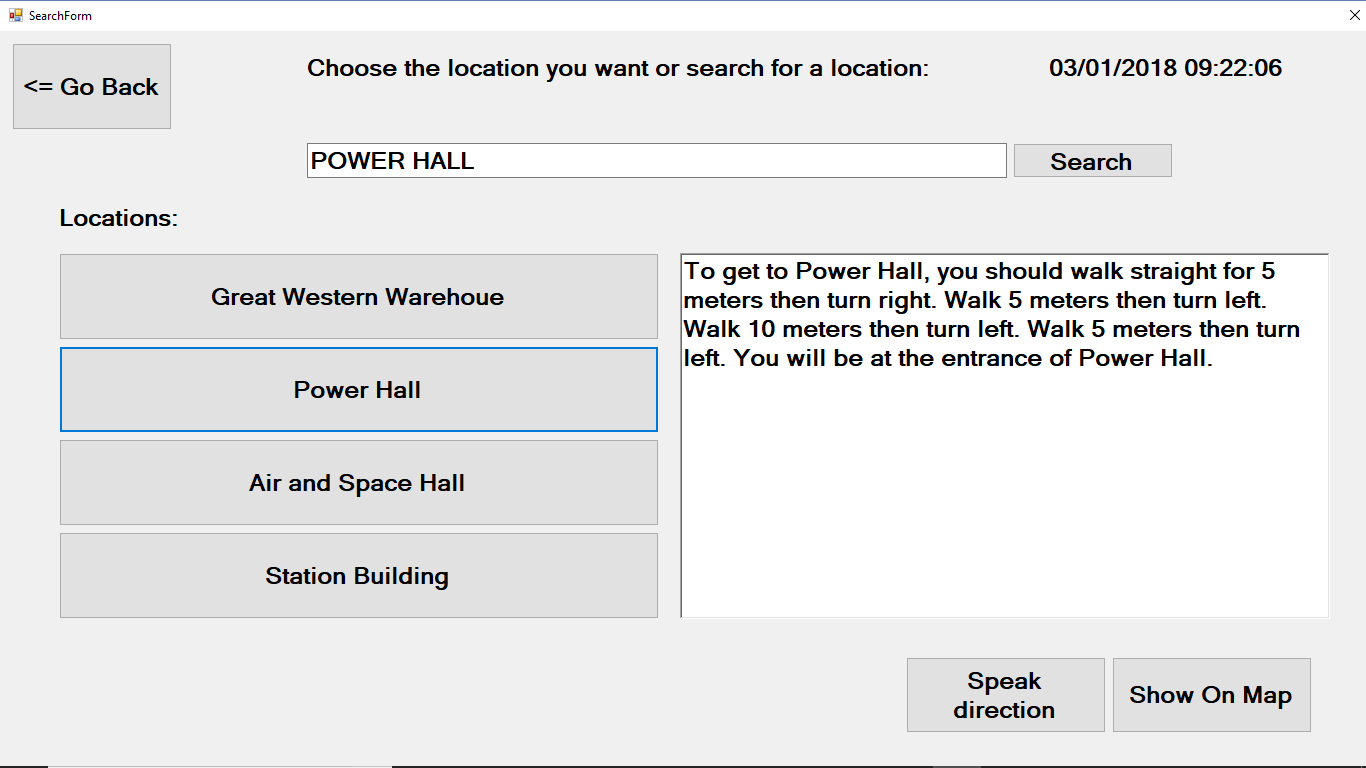
(4.3, 4.3.1, 4.3.2 and 4.3.3)

This form shows what happens when the user clicks on the search bar. A keyboard appears under the search bar which allows the user to type in a location. The keyboard allows for characters to be enter by clicking on the character buttons just as seen in the screen and allows for spaces by using the space button. Clicking on the delete button will also remove one character (most recent one).



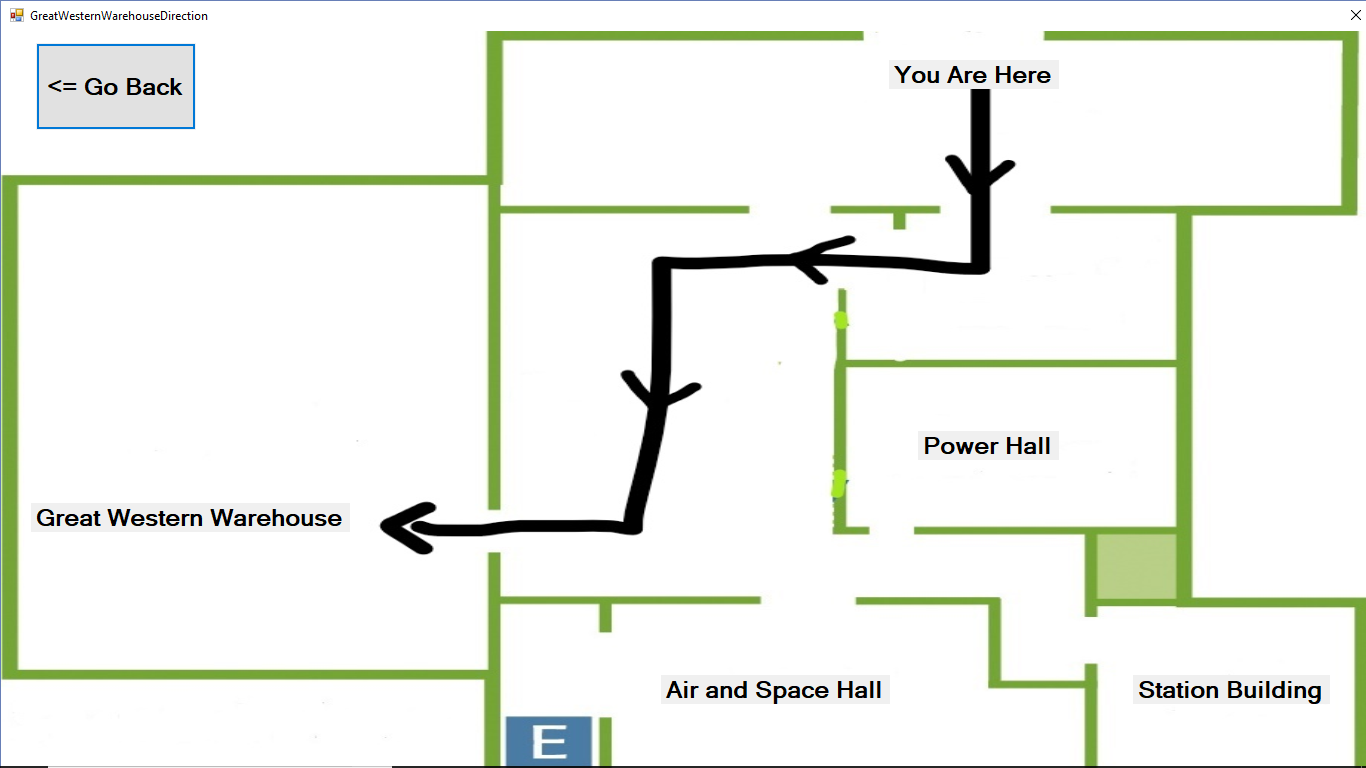
(4.2 and 4.3)

If the search bar is used to search for a location, then the textbox will return something. In the screen above, the user types in a non-existing location so the result was not found. The user also had to click on the search button after finishing entering the characters to hide the keyboard and find out the result which is shown in the textbox.



(4.2 and 4.3)

If the user searched for something and it exists (a location), then once the user clicks on search, which causes the keyboard to disappear, the textbox will return a location of the place.



(4.5)

If the user searches for an existing location or clicks one of the given locations, and the textbox gives feedback telling the user where the place is, then the user has an option to click on the show on map button to see a visual image of where he is and how to get to where he wants to get to.

**4. Evidence of user testing (at least 5 users) with conclusions and brief summary of subsequent   
design modifications and recommendations:**

10 people will be testing my programme to see if the programme is functioning the way it is supposed to. There are 3 objectives to be meet in this test which are; effectiveness, efficiency, and satisfaction level.

When it comes to effectiveness, I will be testing two things which are; completion of task and accuracy (how many mistakes the user made). In terms of efficiency, I will be testing how long it took the user to complete the test. However, satisfaction level is very hard to measure since it’s subjective to the user. Therefore, I will be asking each user to rate (out of 10) how easy it was for them to use the programme, in other words how easy it was for them to complete the tasks given to them.

Before explaining to the user what they should do, I will be giving them a little background information about the programme and the reason it was built so they can have a better understanding of the tasks. I will then be giving each user 3 tasks to complete which are; to sign up to any workshop they want, to search for a location using a faceted search (no search bar used), and to search for the location “power hall” using a text search.

To make my test as reliable as possible, I will be making sure that the tests are done in the same environment. Another thing that I will do to ensure that my results are accurate is to choose a wide range of users. My goal is to pick 5 males and 5 females, and to have at least 1 user from each age category. The categories that I will be using are; 0-8, 9-16, 17-24, and 24+.

**Results:**

Task 1: Sign up to any workshop.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User | Gender | age | Time taken (s) | Mistakes | Task completion | Level of difficulty |
| 1: Nasra Hagi | female | 24+ | 193 | 1 | Yes | 6 |
| 2: Assura | Female | 17-24 | 48 | 0 | Yes | 1 |
| 3: Hamda Musse | Female | 0-8 | 124 | 0 | Yes | 3 |
| 4: Zainab Hashi | Female | 9-16 | 84 | 0 | Yes | 0 |
| 5: Safiya Jama | Female | 17-24 | 109 | 1 | Yes | 2 |
| 6: Hafsa | Female | 17-24 | 81 | 0 | Yes | 0 |
| 7: Hashem | Male | 17-24 | 48 | 0 | Yes | 0 |
| 8: Abdi Jama | Male | 24+ | 140 | 1 | Yes | 4 |
| 9: Bashir Ahmed | Male | 9-16 | 61 | 0 | Yes | 1 |
| 10: Nuur Abdi-Salam | Male | 0-8 | 144 | 0 | Yes | 3 |

Task 2: search for a location using a faceted search (no search bar used).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User | Gender | age | Time taken (s) | Mistakes | Task completion: | Level of difficulty: |
| 1: Nasra Hagi | female | 24+ | 59 | 1 | Yes | 3 |
| 2: Assura | Female | 17-24 | 29 | 0 | Yes | 1 |
| 3: Hamda Musse | Female | 0-8 | 58 | 1 | Yes | 4 |
| 4: Zainab Hashi | Female | 9-16 | 49 | 0 | Yes | 5 |
| 5: Safiya Jama | Female | 17-24 | 57 | 0 | Yes | 3 |
| 6: Hafsa | Female | 17-24 | 14 | 0 | Yes | 0 |
| 7: Hashem | Male | 17-24 | 74 | 1 | Yes | 5 |
| 8: Abdi Jama | Male | 24+ | 53 | 1 | Yes | 3 |
| 9: Bashir Ahmed | Male | 9-16 | 35 | 0 | Yes | 2 |
| 10: Nuur Abdi-Salam | Male | 0-8 | 44 | 1 | Yes | 4 |

Task 3: to search for the location “power hall” using a text search

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User | Gender | age | Time taken (s) | Mistakes | Task completion: | Level of difficulty: |
| 1: Nasra Hagi | female | 24+ | 138 | 3 | Yes | 6 |
| 2: Assura | Female | 17-24 | 31 | 0 | Yes | 1 |
| 3: Hamda Musse | Female | 0-8 | 95 | 3 | Yes | 4 |
| 4: Zainab Hashi | Female | 9-16 | 62 | 1 | Yes | 2 |
| 5: Safiya Jama | Female | 17-24 | 57 | 0 | Yes | 0 |
| 6: Hafsa | Female | 17-24 | 46 | 1 | Yes | 3 |
| 7: Hashem | Male | 17-24 | 67 | 1 | Yes | 6 |
| 8: Abdi Jama | Male | 24+ | 85 | 2 | Yes | 2 |
| 9: Bashir Ahmed | Male | 9-16 | 60 | 1 | Yes | 3 |
| 10: Nuur Abdi-Salam | Male | 0-8 | 86 | 1 | Yes | 3 |

**Calculations:**

Average time taken in task 1: 103.2 seconds

Average time taken in task 2: 47.2 seconds

Average time taken in task 3: 72.7 seconds

Average mistake made in task 1: 0.3

Average mistake made in task 2: 0.5

Average mistake made in task 3: 1.3

Average level of difficulty in task 1: 2 out of 10

Average level of difficulty in task 2: 3 out of 10

Average level of difficulty in task 3: 3 out of 10

**Conclusion:**

From my results, I can see that all 10 users have completed each task given to them which shows that the information in the programme was sufficient enough to guide the user through the interface.

When I look at task 1, I can see that, in average, it took the longest amount of time to complete which is understandable since it involves many steps. However, it seems that this is where users made the least amount of mistake where only 3 people made a mistake when typing in their name using the touchscreen keyboard. It also turned out that the users found this task to be the easiest to complete out of the three tasks.

On the other hand, task 2 turned out to be the quickest task to be completed by the user, which only took them an average of 47 seconds. This is again expected since it is the shortest task out of the 3 and therefore would require the least amount of time. However, when compared to the first task, the users made more mistakes in this task. This was possibly due to one of two reasons. The first was that the users instinctively went to the search bar to choose a location which shows that they either didn’t read the information on the top of the page or didn’t understand it. The second reason is because the task wasn’t clear enough for the user, so they didn’t understand what was required of them. The users also found this task to be more difficult than the first task.

The third task took an average of 73 seconds to be completed. However, this is where the users made the most number of mistakes. some users were looking for an enter button to click after typing in “power hall” and didn’t know that they were supposed to click on the search button. Others, decided to go straight to the show on map button after typing “power hall” without clicking on the search button. Some users couldn’t see the search bar when the keyboard appeared and had a problem with that while others didn’t mind or seem to notice. However, even though users were making the most mistakes in this task, they seemed to find it, on average, as difficult as the second task where the least number of mistakes were made which seems strange.

Overall, I believe that these results are not enough to come to a definite conclusion and would require more testing and results. However, from what I have gathered, it seems that the programme has completed its task in guiding the user through the interface as shown in my results.

**Modifications:**

To ensure that the results were more accurate we could’ve had 1 participant a day where they do the test at the same time and in the same place under the same conditions. This would’ve reduced the number of factors affecting the result which would ensure that the results were more accurate.

There are however some modifications needed if I were to make the programme itself more efficient and make it easier for the user to use. The first thing that I would change is the layout of the name Form. Some users weren’t sure if they were supposed to fill in their full name or only their first name. Therefore, a modification that I would make would be to have 2 textboxes in the same form (1 for first name and another for surname).

Another thing that I would change would be the location of the message pop-up box that tells the user they have registered to a workshop. The placement of the pop-up box is random and appears in different places for different users. I would change this by making the default location of the pop-up message form in the centre of the screen to capture the users’ attention, so they can read it. Another similar problem was that when the user was doing task 3, the keyboard would hide the search bar which would cause some problems when they wanted to see what they typed. Some fixed this by dragging the keyboard location under the search bar. However, a modification I should make would be to set the location of the keyboard to be under the search bar and lock it there so the user won’t be able to move it around freely.

When it came to task 2 and 3 of the testing, some the users didn’t press on the option for speech. However, for those that did complained that the once the speech option was clicked on, they couldn’t do anything else until the speech was complete. This could be a problem if the user decides to not search for a location and presses on the back button. This action won’t take place until the speech is over. Therefore, a modification that I would make would be to allow the user to stop the speech at any time by either clicking the back button or clicking/searching for a different location.

The final thing that I would like to change, or more specifically add, is an option to change the volume of the speech. Keeping in mind that this programme will be in a public space where there will be other people talking and disturbances around the user, we should give the user an option to increase the volume of the speech to make sure that they can hear what is being said.