

Assignment P2

Gaurav Pande

gpande@gatech.edu

Question 1

In this Question we will look into the course registration process for the online student using the OSCAR tool built by Georgia Tech and discuss the potential design challenges and solutions.

The Process:

Currently a student register for a course using the OSCAR tool. Here are the brief steps if someone have to register for the course(there can be another ways as well, but this is the standard one shared during orientations for semesters):

1. Login to Buzzport[*click 1*].
2. Click on OSCAR link (this can be found under the Registration and Student Services panel on the right side)[*click 2*].
3. Select Student Services and Financial Aid[*click 3*].
4. Select Registration[*click 4*].
5. Select Registration Status[*click 5*].
6. Select Look up for classes[*click 6*].
7. Do an Advanced search based on online student[*click 7*].
8. Select a course there and Register[*click 8*].

The Problem:

The problem here is that the current process involve a huge gulf of execution and cognitive load on a student to register for a particular course. First there are almost 8 clicks to reach to choose the subjects, than in order to search for a course you have to be very careful for choosing the filters to get the required courses(the most tedious and cognitive load task here on advance search is scrolling down and choosing your department as computer science and computation science as shown in Figure 1).

Subject:	<div> Accounting Aerospace Engineering Air Force Aerospace Studies Applied Physiology Applied Systems Engineering Arabic Architecture Biology Biomed Engr/Joint Emory PKU Biomedical Engineering </div>					
Course Number:	<input type="text"/>					
Title:	<input type="text"/>					
Schedule Type:	<div> All Co-op Work Assignment Directed Study* </div>					
Credit Range:	<input type="text"/>	hours to	<input type="text"/>	hours		
Campus:	<div> All Armstrong Atlantic/GTREP Foreign Exchange </div>					
Part of Term:	<div> All First Half Term Full Term - Drops Prohibited </div>					
Non-date based courses only						
Instructor:	<div> All Abdelfattah, Heba Abdel-Khalik, Said I </div>					
Attribute Type:	<div> All Humanities Requirement Low-cost \$40 or undr req txtbk </div>					
Start Time:	Hour	<input type="text" value="00"/>	Minute	<input type="text" value="00"/>	am/pm	<input type="text" value="am"/>
End Time:	Hour	<input type="text" value="00"/>	Minute	<input type="text" value="00"/>	am/pm	<input type="text" value="am"/>
Days:	<input type="checkbox"/> Mon <input type="checkbox"/> Tue <input type="checkbox"/> Wed <input type="checkbox"/> Thur <input type="checkbox"/> Fri <input type="checkbox"/> Sat <input type="checkbox"/> Sun					

Figure 1 : Advance search tool to search for a course.

The Solution:

One way I can compare the online registration is with the online e commerce site like amazon.com. Just like we purchase different products and see their availabilities, their feedbacks, similar is the process of registration for a course as well. You look for the courses(like products), than you look if they are available for you to register, i.e waitlisted or not, and then you need a mechanism to get the feedback for that particular course and finally you pay using your Bursar account. Direct manipulation aims to remove the need of an interface, rather it focuses on the direct interaction with the task as defined in the model-world metaphor. There is a direct engagement of the task there. The best way to achieve that here in the course registration process is to separate the registration process from the buzzport and provide a direct interface where once a student login with his gatech id, he should be able to directly see all the course for that semester (If the registration period is over he should be

greeted with the courses he has registered or done before). The courses in that interfaces should be treated like objects like we see in the any e commerce site where we can see the availability of the product(from visual perspective). In that way there will be a direct interaction with the course a student wants to take. We have to be careful here in designing the course page as well because in the current portal its a table having all the courses with various jargon(to understand which each student would need to be trained and that should not be the purpose of our design). **One** of the biggest advantage of this design is we will reduce the gulf of execution by 8 clicks to 1 click, and secondly the interface will be able to better describes the knowledge and needs as it will directly show all the courses the student already registered(can simply even drop or withdraw) and there can be nice feedback options with stars just like we have in amazon.com for product review because in online platform it is important to know the reviews and based on that a student can select the course, put it in wishlist for future semester(that way the management know a rough estimate of load for next semester) and simple add to the cart(meaning register the course easily with a 2 clicks). Last but not the least you can integrate your bursar account payment through same platform. These are all the reducing the gulf of executions and thereby involving the users directly with the task of buying/registering the courses. **Secondly**, there is direct interaction of a student with the course, it is like mimicking the same process the way we purchase a product online. You don't have to worry about any additional clicks or anything, and people have practice to user ecommerce site, so it is much closer to invisible interface as they know the design already.

Question 2

In terms of invisible interfaces qwerty keyboard is one of the thing which I guess bugged everyone with a lot of effort and practice to get acquainted with it. Initially when I had started using the qwerty keyboards, it was almost very difficult to type even a single letter, it is like I have to search my eyes all over the keyboard layout to just find a single character, and one can imagine how hard it was to write a 1 page at that time. But as the time pass by I get used to the layout and now the fingers have learned the layout and can easily type the words. It is now like invisible to me as if I am typing, if i need some words to be written on the screen my fingers can do that

automatically without knowing it is interacting with the keyboard. But I can see the agony in the face of someone new to computer or keyboard.

The alternate solution to this pain of learning of typing on a standard qwerty keyboard is to remove the keyboard itself. It may sound funny at one point but with the emergence of the voice based command and typing I think we will be less using our hands while typing in future. So why not instead of using keyboard we start using voice based typing and start training our voice models to learn from our command. I can understand that one of the challenge initially would be that it will be harder for the voice based interface(alexa or whatever) to learn the voice and accent and the typical demographic slangs and synonyms(there are lot of issue with alexa and other voice interfaces), but with machine learning the voice model can be trained in much less time, thereby reducing the gulf of execution and have a more direct interaction with the interface. There could be buttons like on our computers like one to train our voice model and other to test, that way we can completely remove the dependency on the finger or touch based typing. The training could be done via auditory feedback mechanism. Let's take an example, suppose you ask voice model "add breads to my shopping list", the voice model which currently have groceries list keyword in its trained model can suggest the user as "did you mean groceries list?". In a similar manner we can devise various train mechanism to train the voice model. We can even train with slangs as well. Once we train the voice model based on our voice and input, it would immediately be available to follow your commands and convert speech to text. The biggest advantage of this would be when a voice model gets matured to a particular demographic area, it will be extremely useful for novices to use the voice based typing.

Question 3

Smart watch to navigate pedestrian

In most of the android based smart watches the way to navigate the pedestrian is integrated with the google maps and the feedback to user when someone is navigating is very important. The answer to this question is based on my experience

on the LG smart watch. The features in the new watches might be new and advanced.

Visual:

In a smart watch since the dial of the watch is limited in area the visual feedback receive is also limited and most of the time when a person wearing a watch and navigating does not expect to watch the screen all the time but it becomes important when you are navigating in the dark. Currently, when a user starts the navigation there is not much of a visual feedback given to a user, however whenever you touch the watch the screens lits and it shows your position.

One of the way in which visual feedback can improve the user experience while navigation is to alert the user at important milestones, i.e whenever you going for a turn while navigation the light of the watch lits and shows you the correct turn to avoid taking a wrong turn. This can be especially helpful when you navigate during night.

Auditory:

There is no auditory feedback on the smart watches, however there are auditory feedbacks in google maps if you use it on the phone. I guess that is partially because it is an extra cost to add speaker on a watch which add another design paradigm for the watch. Today the navigation systems on the most of the modern smart watches don't rely on the third party apps like google maps(which needs to be integrated or connected with your smart watch), rather you can navigate simply using the GPS.

Having an auditory feedback can definitely improve the user experience but it might compromise other design feature of the watch(like the design now might look bigger due to addition of a speaker or the cost of the watch become larger and other factor but we can ignore those in this context). One way the auditory feedback can help is by constantly giving the directions to the user in a auditory manner, so whenever a user navigate he doesn't have to rely on the visual feedback always and can navigate using the auditory feedback.

Haptic:

The most common form of haptic feedback that can be given to a user is using the “vibrations” of a watch. Currently the way haptic feedback works in a smart watch is whenever you reaches a critical decisive point while navigating like taking a left turn or right turn. In the current watch that I uses the watch vibrates whenever I reaches a turning point and whenever I reached my destination. This is very important as sometimes it may happen that a user might forget and mislead to a wrong path and the haptic feedback can alert the user to watch the navigation and look for the auditory or visual feedback.

The haptic feedback can be improved if the alerts are given for various other events like whenever there is red light the watch vibrates and alert the user to stop. It can also vibrate if somehow a different optimized route is calculated based on the traffic congestion scenario. Such integration and feedback can definitely help the user experience.

One of the other perception that I feel can be useful while navigating a router using smart phone is using the pain perception or Nociception, which can be useful in a situation where a navigator is about to hit by a bus or vehicle or any falling building from above. Obviously this would require our navigation system to be smart enough to detect the objects, their location and their speed in order to alert the user in a situation when a vehicle might collide and can alert the user with a pinch or pain to tell to run or jump.

Question 4

Tip1: Emphasizing the essential content and minimizing clutter

One of the interface that I use frequently is piazza and I honestly doesn't like the functionality of having everything on one page philosophy of piazza. The first time I

logged into Piazza, I had no idea where I was supposed to focus, or what I was supposed to do. There is so much to do in Piazza, but most of it does not interest me. I am typically short on time, so unless I am having an issue, or looking to have a discussion, the main reason I log on to Piazza is to see posts from the professor. But it is so hard to look for the post of my interest that most of the time I get lost into the myriad of posts line by line and I have to keep on scrolling down. Although there is an option to select the posts posted by instructor but the interface put a high cognitive load that you need a training to learn to get accustomed with the interface.

Therefore, I would like to design the interface of the piazza not be based on “everything” on one page, rather I would like to have a widget based homepage where a user have a choice to look for the post he is interested into. If I am interested into only the post of Prof A, then I should be able to view only the post of prof A. If I wanted to see the most viewed question then I should be able to do that without much of a cognitive load. So in my design the first time the user logs in to piazza, the system would ask the user what type of post or what are his interest and then accordingly the post are visualized to him. This would allow me to see only the information I am interested in, thereby decreasing a high amount of cognitive load.

Tip2: Give the user control of the pace.

One of most used app while pursuing online masters is using udacity for various courses, however the currently functionality of autoplay of the next video in the next 5 second adds a cognitive load on a user. What if I don't want to view next video? What if I want to write down some notes? What if I want to review the video? I can't do this until I have the control of the autoplay next video. The current feature for next video play within 5 second put a cognitive load on the mind to decide in between what to do next. Shall I go with next video? But wait did I understand the current video? By the time you think this the next video starts up!

So by giving the user to control the task by controlling the pace of the task I would design an interface where we can just ask the user before starting a chapter what is

the wait time he would like to have in between 2 video. It could be a plain text box or a dropdown as well. Based on this a user then can start his chapter without worrying about the cognitive load and can easily revise or make notes in between 2 consecutive videos.