

# Assignment P2

Kyla Qi  
kichy4@gatech.edu

## 1 QUESTION 1 (FROM LESSON 2.3): ~1.5 PAGES

The five tasks I chose were: texting my friends, locking my apartment door, using an app to enter a virtual queue for (socially distanced) dinner with my (vaccinated) friends, ordering via an app, and paying for said food with the same app. I've created a table below to explain these tasks' associated goals, interfaces, and objects.

*Table 1*—The five tasks I chose and their corresponding goal, interfaces, and objects

Task	Goal	Interface(s)	Object(s)
Texting my friends	Communicating with a few people simultaneously	Signal app, keyboard	the text, my friends
Locking my apartment	securing my things in my apartment	lock, keys	door, bolt
Joining a virtual queue	getting an open table at bartaco	queuing app	the queue
Ordering food	communicating what food I'd like to eat	the ordering app, my camera (to scan a qr code)	the food
Paying for food	completing a transaction	the ordering app	money/credit card

### 1.1 Level of Directness and Invisibility

Texting my friends is not a very direct task - pressing buttons displayed on a glass screen with no haptic or aural feedback is not an instinctive way to write or communicate information. However, the interface is practically invisible to me, which was due to a combination of both the intuitive user interface of Signal and the many years of practice I've had with using a mobile keyboard.

Locking my apartment is a very direct activity; since I have a regular lock and key instead of some kind of fob scanner, I can feel and hear the bolt sliding into place. On the other hand, the interface isn't entirely invisible. Using a key and lock is a common exercise, but for some reason I can never remember which way

to turn the key in order to lock the door. Thus I often have to spend time twisting the key back and forth until I hear the clicking sound that indicates that the bolt was moved into the locking position. In this way I spend more time thinking about the interface than I do the actual task.

Ordering food at bartaco was not a direct nor was it an invisible experience. The page I used to order my food was accessed via a QR code and our food selections were made by clicking through the menu. This is very far from directly manipulating the objects behind the task, though it's slightly more direct than giving the waitstaff your order, as the app had a visual representation of the food. The interface itself was buggy and slow, and the QR code scanner on my phone took a while to locate the correct webpage. There were also several instances where an accidental gesture caused me to navigate to a different page of the menu that I wasn't interested in.

Paying the bill was similarly beleaguered. Since money is really just a concept it's hard to quantify what would count as a more direct way to interact with it via the app, but since everything was done via clicking buttons and entering payment information the experience could definitely be *more* direct. The interface was also not very invisible; this was my first time using the application and the presence of options to split the bill and preemptively enter payment information resulted in my spending a lot of time trying to understand the interface instead of focusing on the task at hand.

## **2 QUESTION 2 (FROM LESSON 2.3): ~1.5 PAGES**

Krita is an open source raster graphics editor (aka a drawing application) that is used by professional and amateur artists alike. Drawing software often has to support a lot of operations while also making them efficient to access. This often means that the most frequently used tools will be easy to locate, but less frequently accessed tools will be harder to find and also require more clicks to access. Furthermore, since artists often use a tablet to draw, having to move their pen across the entire screen to adjust settings while in the middle of working can be intensely frustrating. Some major components I struggled with when I started using Krita were efficiently changing my brush type and size, changing the color I was using, and changing layers. Most of these functions were easily accessible from the main interface, but having to switch between selecting functions and working on the canvas made the drawing process slow and unpleasant.

Furthermore, there were obscure tools that I had grown accustomed to accessing using SAI, a drawing application I had used before swapping to Krita, which I now struggled to access.

After a lot of learning, I don't have to spend nearly as much time thinking about Krita's interface. This is thanks to a combination of factors, the most important of which were the Krita specific hotkeys and the location of more obscure tools that I discovered by searching user forums and reading documentation. Krita also offers a decent amount of personalization of the interface; their ability to favorite certain brush types means I can now access my most used brushes without having to scroll for them. I've added my own custom hotkeys that match the shortcuts on SAI, so I wouldn't have to completely modify my behavior. I also enhanced the efficiency a step further, mapping my most used hotkeys to buttons on my drawing tablet. Now, for certain actions I don't even have to stop to find the right key on the keyboard, I can simply press one of the four buttons on my tablet or even one of the buttons on my pen. The combination of easier access and muscle memory from repeatedly performing the same tasks on the Krita interface means that I now can focus on the canvas instead of the tool selection interface.

The most impactful enhancement I think could be added to improve the learnability of the interface would be displaying the keyboard shortcut of a tool in its mouseover alt text. Currently, when a user mouses over a tool, the screen only displays the name of the tool. It would greatly speed up the learning process if the shortcut information was easily accessible via the interface instead of requiring that the user go searching for it. Another feature that could speed up the process of reaching invisibility would be adding graphics to the keyboard shortcut menu. Though most tools have a name in their alt text, many tools are only represented by an image on the main interface, so once the user tries to configure a shortcut for a tool it can be hard to identify what their corresponding name is in the shortcut configuration menu.

### **3 QUESTION 3 (FROM LESSON 2.4): ~1.5 PAGES.**

I selected the task of "Cooking a meal, including using the various appliances and utensils".

### **3.1 The current state of user feedback**

All three common types of human perceptions are relied upon heavily throughout the food preparation process. When cutting up raw ingredients, visual feedback tells the user if their cuts are an appropriate size. When preparing dough, haptic feedback tells the user if the dough has developed enough gluten (based on how tough it is to knead), and if it has the appropriate level of stickiness.

During the cooking process, visual feedback and audio feedback tend to complement each other - the sizzle of frying food and the bubbling of boiling water are also accompanied by visual feedback. In some cases, however, haptic feedback is more helpful - this is often the case with determining if the ingredients are cooked thoroughly. I will often sample some food (excluding meat) from the pot before I turn the heat off; the mouthfeel gives me the feedback I need as a user to determine if the food has been cooked to the desired level of doneness. Visual feedback can also be useful in this case - the color of a baked good or the perceived stiffness of noodles or the browning of meat can indicate to the user it's time to start checking the food for doneness.

### **3.2 Other possible ways to give feedback via vision, sound, and haptics**

A visual indicator of the temperature of the stovetop would be very helpful to users, especially for those who have electric (and therefore inconsistent) stoves such as myself. Without knowing exactly the level of heat being applied to the stove top, the user is forced to infer the level of heat being applied from the feedback they received from the pot's contents.

An audio alert would let the user know that their pot has reached boiling or is about to boil over would be useful for users that have multiple tasks going on in the same environment and need to pay attention to other tasks while they wait for water to boil. I personally have had a few incidents with pasta that boiled over the side of my pot because I was chopping vegetables with my back facing the pot. An audio alarm would be impossible to filter out and alert the user.

Similarly, haptic alarms (via a smart device) would be useful in cases where the user is preparing food for a social function and doesn't want to disturb their guests when a dish has finished baking or cooking. This is different from timers with a silent alarm since the alarm would be based on the doneness of the food.

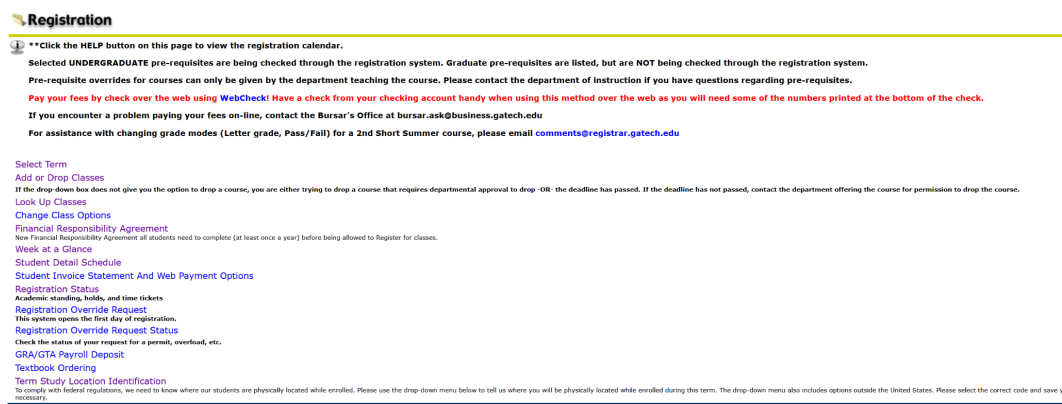
### 3.3 Another type of human perception

Olfactory perception is arguably the fourth most important type of human perception for cooking. Smell can give us feedback on the ripeness or spoilage of raw ingredients, clue us in on how food might taste without having to sample it, and let us know that something is burning before we see it or set off the smoke detector alarm.

#### 4 QUESTION 4 (FROM LESSON 2.4): ~1.5 PAGES

##### 4.1 Emphasize essential content while minimizing clutter

The OSCAR interface does not do a good job of emphasizing essential content. All functions on the main menu are the same size, font, and color, though there are some functions that should obviously be emphasized, like the “Lookup” Classes” function. Similarly the course section listings treat irrelevant course codes with the same formatting that they do for more relevant information like the lecture timeslot and number of available seats.



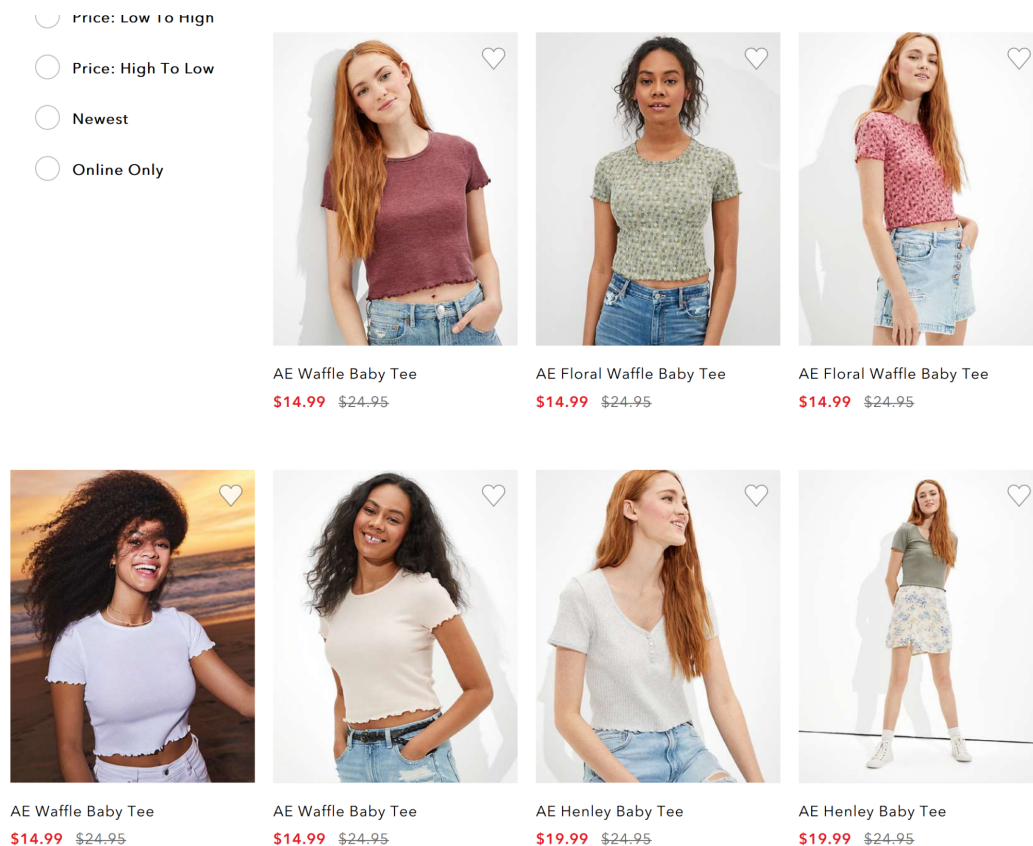
*Figure 1*—The OSCAR main page. Some links are purple instead of blue because the screen was captured after the user had already clicked some links.

If I were to improve the main menu of the OSCAR interface, I would use a combination of grouping together pages and removing pages according to the user as a predictor model. Course selection options (“Add or Drop Classes”, “Look Up Classes”), for example, would be emphasized with larger text and more eye-catching formatting. “Student Detail Schedule” and “Week at a Glance” would go under one “View Current Schedule” option. The term selection option would become a drop down menu on the upper right hand corner of the screen,

where the currently selected term would also be displayed. I would move the Financial Responsibility and Term Study Location Identification interfaces to a single “Manage/View your Holds” page. Similarly, the two registration override pages would be grouped into one page. I would also take advantage of the entire screen, instead of putting every option in a list on the left side of the page. Finally, I would remove the Student Invoice, Payroll, and Textbook Ordering from the page. While those options are important and useful I don’t believe it makes sense for them to be on the same page as course registration.

#### 4.2 Offload tasks from the user onto the interface

The American Eagle online shopping interface is generally decent. However, one of the things they do to increase the amount of work the user has to do is in the way they organize their garments for browsing.



*Figure 2*—A sample of AE’s wares under the “Women’s Shirts” category.

As can be observed in the above image, American Eagle lists each individual color and print for the same cut of shirt on the browsing page. Not only this, but they do not group them in a logical fashion; the Floral Waffle Baby Tees are listed before the rest of the Waffle Baby Tees. Since the user could just mouse over a shirt and click on the "Quick Shop" option to view other available colors, the extra listings only cause the user to scroll more. Further exploration of the website would also reveal that garments are grouped by the cut *and* the print, meaning the solid color Waffle Tees would be on one page while the floral Waffle Tees would be on another. This results in the user having to open extra pages and do more clicking around to view all of their options, giving them more unnecessary tasks while they're already trying to decide what to purchase.

To redesign this interface, I would group all of the garments with the same cut into one listing on the main browsing page. The detail page of the garment would include every color of the garment, including patterns. This would remove the burden of grouping together garments of the same cut from the user. It would also reduce the amount of scrolling the user has to do by virtue of drastically reducing the length of the main browsing page.