Assignment P2

Stephen Freeman
sfreeman66@gatech.edu

1 QUESTION 1

Table 1 — Tasks accomplished in a one-hour period

Task	Goal	Interface	Object
Using a manual coffee grinder	Make coffee grounds for a cup of coffee	coffee grinder lever (manual), jar that twists open and close, coarse setting dial	Coffee
Using an electric kettle	Hot water to brew coffee	Electric Kettle's dial, lid, and gooseneck	Water
Using a washing machine	Having clean clothes	Dial for cycle, dial for water level, dial for temperature	Clothes
Using a microwave	Heating up food	Microwave button controls, microwave handle	Food
Navigating YouTube on the phone	Watch videos	YouTube search bar, YouTube results list, YouTube video player	Videos

1.1 Coffee grinder's level of directness and invisibility of interaction

Because the coffee grinder is a manual hand grinder, the user is very close to the actual manipulation of the coffee into grounds. The user only has one chamber in which coffee can be inserted, and once the hand crank is on, the user only has two directions they can turn the handle. The correct way will have immediate auditory and haptic feedback as the user can hear and feel the grinders grinding the coffee as they turn the handle. The grinds collect at a jar on the bottom, which immediately appears as the user interacts with the interface. If this were an automatic grinder with a motor, there would be another layer of indirection, as the motor would be doing the actual manipulation of the coffee.

For the invisibility of the interaction, the only thing the user must be cognizant of is the coarse setting. This is the only interaction with the interface that does not go towards the actual manipulation of the coffee and something that does not make the interface invisible. Once the user is aware of this setting, however, the rest of the interface is basically invisible as the user is only focused on cranking the handle to reach their goal.

1.2 Electric kettle's level of directness and invisibility of interaction

For the Kettle's level of directness, there is a difference between its input and output. For the input, the kettle's temperature is handled by a dial. Between the user and the water, there is a dial to set the temperature, the heating element, and the kettle holding the water. Once the water is up to temperature, the user controls the kettle directly to output the water onto the coffee grounds. The gooseneck allows for fine control, so the user just directly manipulates the tilt of the kettle to control the water flow directly.

For the invisibility of the interaction, the only decision the user must make before directly interacting with the water is to turn the kettle on and set the temperature via the dial. Once that interaction is complete, the user simply waits until the water is at the temperature they need, and they can focus on the pour rate of the kettle for their pour over coffee. The kettle almost becomes invisible at this point because of the precision the gooseneck provides.

1.3 Washing machine's level of directness and invisibility of interaction

For the level of directness with the washing machine, the user is far from the actual manipulation of the object. The user sets the cycle, water, and temperature and then leaves the interface alone until the task has been done. Even though the user is not directly manipulating the object, it allows them freedom to do other tasks in the meantime.

For the invisibility of the interaction, unless the user groups their clothing by fabric or settings, the user must always think about which settings to use for their piles of clothes. Novices will interact more with the interface at first than with the task. As the user becomes more familiar with the device, they can gain expertise and quickly go through the interface to almost make it seem like it is invisible, but the factor is expertise.

1.4 Mobile YouTube's level of directness and invisibility of interaction

For the level of directness with YouTube, the user feels like they are directly manipulating the object of the task. When the user wants to use the search bar, its intuitive to just click on it, bringing up the user's keyboard to type in what they want to watch. The search results are similar in that the swiping up and down the list will intuitively move the results and touching a result will bring up the video. Just touching the video will bring up the seek bar and controls such as pause, rewind or next video.

Because the level of directness is proportional to the cognition it takes to use the system, the interface almost becomes invisible. The user, even a novice, can intuitively figure out how to search and find a video to watch. The good design makes the interface feel like it is invisible, where if the user uses it more, it can discover more advanced features by learning. YouTube will describe features when the user touches a video so as the user uses the interface more, they can become more proficient, thus leading to more of an invisible interface.

2 QUESTION 2

A task that I do on a regular basis that has become invisible by learning is interacting with my debug menu for the game I work on at work.

2.1 Debug interface interaction before learning

Beginner level use of the interface involves interacting with buttons and menu options. The debug menu is a red cog that always shows up in the top right of our screen whenever we are in debug mode. Clicking this icon then shows a popup menu that has different types of options to debug, depending on which field we like to debug. This involves directly manipulating user states, manipulating which stage to go to, manipulating which popups can trigger, and lastly manipulate the game objects in the current scene. Whenever a task needed to be debugged, it involved clicking on the debug menu button, finding the target domain, and then from there picking which fields to manipulate to generate the testable scenario. This involved knowing where each button was laid out and in which group of options it lived in.

2.2 Debug interface interaction now

After becoming more familiar with the codebase, I learned that there are hotkey shortcuts related to the first few letters of each keyword in the current debug context window. Instead of having to place my hand on the mouse (which is not needed in our game) and having to find which order of buttons to press to manipulate a testing scenario, I simply use muscle memory to remember actions I do frequently. An example of this is loading a stage. I would simply press escape, s, and start typing a stage to load, which drastically improves efficiency. The reason for why I no longer spend as much time focusing on the interface is that this interface was designed to teach the user over time (by underlining the hotkey in each word of each option) and by having the debug menu be designed to differentiate for users. New users, like when I first started out in the company, would rely on the menus, but as we would become more proficient and comfortable with putting our project in our long-term memory, we can start to play around with hotkeys with commonly used options.

2.3 Redesigning the debug interface to bypass the menu

A redesign to this interface to get to the point of invisibility more quickly is to have the hotkeys interact with the task specifically itself, and not have the hotkeys interact with the debug menu. Instead of treating the debug menu as the task, we should shift that focus to debugging. We can create hotkeys such as "Control + f" to quickly bring up a search of stages, instead of having to summon up the debug menu with "Escape" and then clicking the hotkey in the debug menu. This lowers the gulf of execution so that efficiency is maximized. This also brings us to interact with debugging faster, rather than interacting with the debug panel, to accomplish debugging.

3 QUESTION 3

Three types of human perception commonly used in user interface design are visual, auditory, and haptic. Exploring an ***advanced exercise bike*** as an example domain, these types of human perception give the user distinct feedback. From their qualities, new designs can be developed. Uncommon senses can also be explored, such as using a human's tension sensors, to give the user feedback as well.

3.1 Current and new visual feedback for an advanced exercise bike

There are multiple ways an advanced exercise bike gives visual feedback to the user. The obvious feedback cues an advanced exercise bike would be to display current biometrics to the user via a screen interface. The screen interface is the quintessential way for the user to receive visual data, as the user's attention is usually centered right onto it. If a user has a target heart rate set, the user can see their heart rate in real time and can adjust, or even have the bike adjust for them, to ensure they are hitting the goals they have set.

The screen can also give feedback of the current lifecycle of the exercise by changing the screen and showing which physical buttons are available at each stage. When the user sits down to start an exercise, the start button can have a built-in lightbulb to start flashing. If a user starts to pedal or presses the start button, the screen can visually display a warming up text and stops the start button from flashing. When the user transitions into the main workout, the buttons for the speed and difficulty to pedal can blink a few times to let the user know that these buttons are now engageable to modify their workout. As the user reaches the end of the workout to cooldown, the screen can flash to a new interface and display a congratulations message, saying the user can now start to bike slowly to finish out their workout.

With a user's visual perception in mind, a new feedback idea would be to create a glass window in front of the user to allow an augmented reality experience to receive a visual interface for feedback. While Virtual Reality helmets have been experimented with exercise bikes, a novel idea would be to have an augmented display so that the user does not have to wear a heavy helmet on their head and worry about getting sweaty in it. The usual visual displays, such as the biometrics would be displayed, but games could even be played where arrows show up and the user must use the steering wheel of the bicycle to go the correct way.

3.2 Current and new auditory feedback for an advanced bike

Many of the auditory feedback that an advanced bike would give the user would pair directly with the visual feedback mentioned earlier. Any button presses or changes in visual cues on the screen interface would warrant a small beep to alert the user that an action has been registered from the user or if a new stage in the exercise has occurred. This fits well with the idea from the lectures to reduce cognitive load, by pairing up two or more modalities to complement each other. These auditory feedback cues are also important, as it bridges the gulf of evaluation. The user can quickly realize that their input they are putting in has been registered or that their exercise (the true task) has started or ended.

Additional auditory cues would be if a user has set any of their biometrics to trigger if they reach outside of a designated range. If a user's goal for their exercise is to be in a certain heart rate, or to reach a certain number of calories burned, the interface can give an extended cue that is different than one from a simple button press to let the user know something different is happening.

A new auditory feedback cue for an advanced bike would be to play music only if the user is pedaling. If the user has headphones in, the bicycle can alert the user to know if they are not hitting a target rate by not playing music as a feedback mechanism. If the user stays within their difficulty range of the workout, the music will continue to play. If the user starts to zone out from their workout, the sudden loss of audio would trigger the user to pay attention.

3.3 Current and new haptic feedback for an advanced bike

For the last common modality, haptics is another source of feedback that pairs well with the other types of perception. If there are no physical buttons on the bike and they are all on the screen, the screen can give a bit of a vibration to let the user know that a touch has been registered. Additionally, when the user goes through the warmup, the actual workout and the cooldown stages, the bicycle can vibrate in the user's handlebars, letting them know that a different stage of the workout has started or ended.

A new haptic feedback cue for an advanced bike could be a system where the bike can interface with a fan to change the temperature the user is feeling. The bicycle can read the user's heart rate to determine how hard they are exercising. If it notices that the user has been at an elevated heart rate, it can turn on or speed up the fan that is pointing at the user. The user will know if they have reached a target heart rate when they notice the fan is starting to blow on them. This can also have the effect of trying to have the user have a more comfortable exercising experience by giving them relief.

3.4 Tension sensor feedback for rehabilitating exercises

Tension sensor feedback is found in the "muscles and allow the brain the ability to monitor muscle tension" (Hiskey, 2016). For users who are in rehabilitation for an injury or have muscle imbalances, they may either have difficulty in using an exercise bike or might not be able to use more of the advanced workouts. If a user thinks an activity might hurt them or is unable to perform a certain action, their body might trigger a defense mechanism to not allow them to hurt themselves.

The advanced bicycle can have their pedals be independent of each other, ensuring that they are still opposite of each other to still mimic pedaling. Because they are independent from each other, each can detect if one leg is stronger or weaker than the other during the warmup phase of the workout. Using this, the bicycle can then set each to have their own relative strength to pedal. The bicycle will continue to adjust to make sure that the user does not negatively trigger their tension sensor. The user will hopefully feel less muscle tension in their weaker leg, allowing them to rehabilitate that leg while still being able to exercise and progressively load to stronger workouts over time.

4 QUESTION 4

There are multiple ways to reduce cognitive load in an interface design. Two of these ways is to ***use multiple modalities*** and to ***emphasize essential content while minimizing clutter***. Two interfaces that violate these principles can be redesigned to help reduce the cognitive load.

4.1 Electric kettle needing multiple modalities

The Stagg EKG electric kettle is a fast-heating kettle with an electric interface. The kettle has one knob and an LCD screen that displays the temperature. The user clicks the knob (which is silent) to turn the kettle on. The user then turns the dial, watching the screen to determine when they reached the targeted temperature to know when to stop. The kettle will then heat up and keep the water inside at the desired temperature once it reaches its goal.

Despite its sleek and easy-to-use design, the only real feedback the kettle's interface gives the user is visual. While the user has the kettle going, the user is probably doing other things to get their coffee or tea ready, such as grinding the coffee beans or getting certain dishes or cups ready. The kettle does not give any auditory or haptic feedback that the desired temperature has been reached. The user must go back to their kettle and look at the screen to see if the target temperature has been reached. There are upgraded versions of the kettle that will alert the user's phone, but a user would have to either pull their phone out or must open it to turn the alarm off. It would just be better if there was an auditory alert so the user knows that they can use the kettle to move to the next part of their coffee or tea routine. A redesign would be for the kettle to perform a small auditory alert, so the user does not have to guess if their kettle has finished reaching temperature.

4.2 Workout apps must emphasize essential content while minimizing clutter

The nSuns workout log app is a workout app available on the Google Playstore that automatically tracks workouts and weights, as well as creates records of previous workouts for a specific workout routine. The main window interface shows the days of the week and by clicking on the day, the workout regimen for that day is shown. The user can edit workouts, weights, timings, and other miscellaneous options in the options dropdown menu.

The application keeps all its options in a singular dropdown menu. While the most essential functionality is in the main window view of the application, trying to edit your workout, weights, program or even your records require looking over an entire list of options. This raises the user's cognitive load, causing them to have to read over an entire list of clutter, where they might even forget what they were trying to accomplish in the first place. A way to fix this would be for the application to group options in their own menu dropdowns based on workout options, program options and recording options.

5 REFERENCES

Hiskey, D. (2016, May 31). *Humans Have a Lot More Than Five Senses*. Today I Found Out. http://www.todayifoundout.com/index.php/2010/07/humans-have-a-lot-more-than-five-senses/