Assignment P3

Piyali Banerjee
pbanerjee32@gatech.edu

1 QUESTION 1

1.1 Design principles' role in creating an invisible interface

Consistency is the design principle which promotes following design conventions that are used across interfaces and that are common and easily recognizable by users. Consistent design allows users to interact with interfaces in ways that are familiar to them as they would have seen similar patterns in other interfaces that contain the same features and functionality, decreasing the gulf of execution, and thus increasing invisibility. For example, by convention, if a user wants to create a new resource in an application, they would have to navigate to the "File" menu which is typically located in the top left corner of the window, and select "New ..." from the drop-down menu. If the new application is designed well and follows that convention, the user will automatically recognize the pattern and will know how to create a new resource -- they will not need to learn a new workflow and the interface will be invisible to them.

Mapping between interfaces and effects in the world is a key design principle that helps create an invisible interface, often by providing visual cues to the user that they are familiar with from experiences in the outside world to make it clear what the effect of an interaction with the interface will be, reducing the cognitive load on the user as well as the gulf of execution considerably, enabling the user to quickly decide on a course of action without thinking too much of the interface. For example, for adjusting the brightness of a screen, the convention is to offer a user control to set the brightness within a range, which is typically represented by icons that the user would be familiar with -- an icon of a sun with short rays for the dim end of the spectrum, and large rays for the bright end of the spectrum. Users are quickly able to make the mapping between the icons and what they signify in the real world, and are able to be confident on the actions they need to take to adjust brightness quickly based on the mapping.

Affordances is the design principle in which interfaces, by design, indicate to a user how to use it, reducing the gulf of execution and thereby increasing the invisibility of the interface. For example, laptops, by physical design, are meant

to be opened by lifting the top to reveal the screen and the keyboard. In Mac laptops, there's a groove in the center of the opening side that is meant to further help the users discern from which side to open the laptop, reducing their gulf of execution.

1.2 Design principles' role in emphasizing participant view of user

Tolerance is a design principle that promotes designing interfaces that are tolerant to user errors by reducing the risk and detrimental effects of such errors. By considering the context in which a user performs tasks and focusing on the participant view of the user, tolerance can be designed in a more purposeful way. For example, in music players such as Spotify, the play and pause icons interchange in the same location of the screen when a user clicks it. One potential motivation for this design choice may be that if a user is in a context in which they have reduced focus and dexterity (such as holding a mobile device while running), they may accidentally click the pause button. Since the pause button turns into a play button at the same location, the user can quickly hit the same spot in the screen to undo their erroneous interaction with the interface.

Constraints are put in place to prevent possibilities of erroneous interactions the users may have with the interface. In order to effectively design constraints, one must consider the participant view of the user by understanding the context in which the user will be performing the action. For example, passengers are able to exit the car door if the door is unlocked. However, if the vehicle is moving, it would be dangerous for a passenger to open the car door. Car designers have considered this safety concern by studying the context, and have designed cars that automatically lock all doors when the car is in motion to prevent passengers from accidentally opening the car doors while the vehicle is moving.

2 QUESTION 2

The alarm clock on my iPhone is intolerant of user errors, as it is easy to accidentally hit "Snooze" when the intent is to stop the alarm. When the alarm goes off in the morning, the first screen I see when I pick up the phone has the name of the time and date at the top of the screen, the name of the alarm in the center, a large bright orange "Snooze" button right under the name, and a light gray "Stop" button at the very bottom of the screen, as shown in Figure 1. Since the "Stop" button is small, matches the background color quite closely, and has "Stop" written on it with a small font, it may be difficult for a user, especially in

the morning when they may be groggy from waking up, to see it. Since the snooze functionality is turned on by default when a user creates a new alarm, they may forget to toggle it off and may mistakenly hit the noticeable bright orange button in the center of the screen, thinking they are stopping the alarm. When the alarm goes off as the snooze time is reached, if they are in a meeting or engaged in an activity that requires full concentration, it would be very distracting and may cause panic for the user as they rush to turn the alarm off.

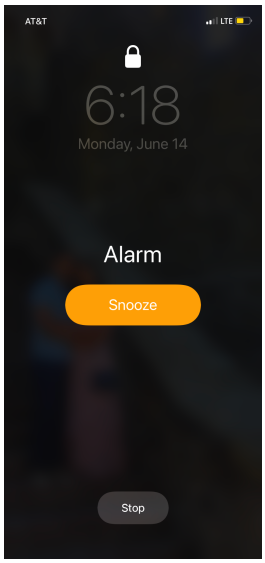

Figure 1—iPhone alarm interface.

One way to remediate the intolerance to the user error is to redesign the interface by introducing a constraint to prevent accidentally snoozing the alarm by adding a confirmation window that pops up when a user clicks "Snooze." This pop-up window can also include an additional piece of information that is not readily available by the interface -- the snooze time. The message may read "Do you wish to snooze for 10 minutes?"

Since the snooze option is more prominent, it may make it difficult for a user to discern that the stop button is a button in the first place (particularly for visually impaired individuals), since it is significantly closer to the background color. An affordance that can be introduced is to modify the color of the "Stop" button to make it more prominent and clear that there is a button on the bottom of the screen.

The alarm interface can further be improved with a mapping by introducing icons to the stop and snooze buttons that the user would be familiar with based on their real world understanding. For example, an icon of a sleeping emoji with a "zzz" would be a clear indicator for snoozing, and a stop-sign icon could indicate stop, to make the effect of each button more clear to the user, reducing the potential for user error.

3 QUESTION 3

The game I selected is Pacman.

3.1 Slip

Pacman is a game that requires a high level of dexterity the player's fingers as it can become fast paced, especially since the pacman moves quite fast and the ghosts seem to become better at finding and cornering the player's pacman as the game progresses. A slip can occur when a player wants to move in a certain direction but clicks the wrong arrow key on accident. On a Mac laptop, the arrow keys are narrow and are placed rather close together, especially the up and down arrow keys. A slip can occur, for example, if a player wants to move up but accidentally taps the down key. One way to change the interface to avoid this is to have one arrow click simply change the direction of the pacman and the subsequent clicks of that arrow causing the pacman to move. Another way to prevent the slip is to provide a better mapping of the arrows to the movements by displaying icons of the arrows on the screen in the console in which the corresponding arrow would light up as the player clicks the arrows on the keyboard so they can better visualize and plan where they need to go and which arrow they want to see light up.

3.2 Mistake

When a new pacman game is started, the pacman appears as a solid, filled yellow circle that takes 3-5 seconds to transform into a pacman with a mouth that is ready to move and eat the pellets on the board. In this pre-start state, a new player may mistakenly start clicking the arrow keys in hopes of moving the pacman but would see that the pacman isn't moving or responding to their arrow clicks. This mistake mainly occurs since the game provides no indication that the game has not started yet and does not give a clear enough indication that the pacman is ready for action. One way the interface can be changed is by

having a message on the screen the moment the player starts the game to indicate that pacman is getting ready such as “Please wait until Pacman is ready” and once the pacman is ready to go, words can flash for a few seconds on the screen to indicate that the player can start, such as “GO!” or “START!”

3.3 Challenge

One thing that makes the game challenging is the speed at which the pacman character moves and the fact that once it’s moving, there’s no way to pause or stop unless the pacman is blocked by a wall or barrier, making it easy to miss turns into the narrow alleys. This is why the game calls for strong dexterity. Another challenge is that the ghosts change colors and has different functions based on their colors -- for example, when they turn blue, they become high-value food for the pacman, so rather than having to run away from the ghosts, the player needs to quickly change their strategy and chase the ghosts instead to maximize their points.

4 QUESTION 4

4.1 Good representation example

The water and ice dispenser in our refrigerator is an example of good representation that efficiently describes what pressing each button would provide with clear icons that light up in a bright blue color when clicked, as shown in Figure 2. Each icon provides a clear indication of what the user can expect will happen when they press the corresponding button. For example, a user can easily understand that pressing the button that says “Light On” on it will turn the light on and the blue light will confirm to them that the setting has been set successfully. This interface **brings the objects (the buttons) and the relationships (icons representing the result of clicking the buttons) together** by placing the icons directly above the corresponding buttons. The interface also **excludes extraneous details** such as the temperature of the ice and water since these are properties the user cannot control via the dispenser.

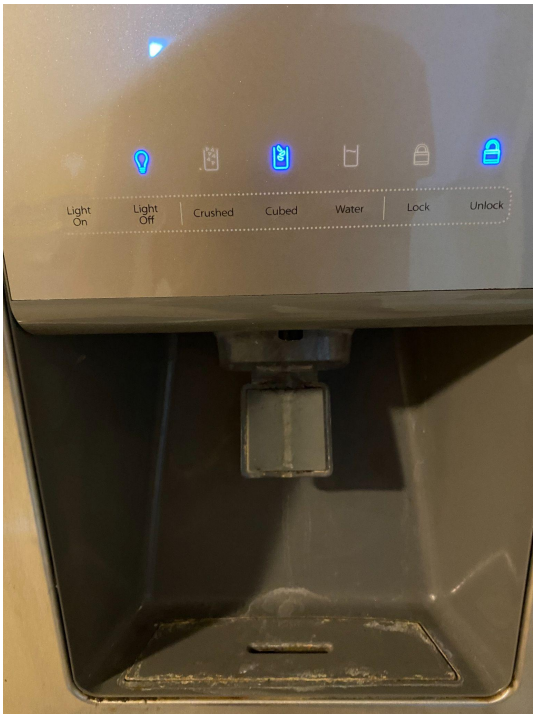

Figure 2—Ice/water dispenser of a refrigerator.

4.2 Bad representation example

A bad representation example is for taking a screenshot of part of the screen on a Mac laptop. Assuming the user has googled to click CMD-Shift-4, all they see come up on their screen initially is their cursor turns into a "+" with the pixel coordinates attached to it, as shown in Figure 3. The pixel coordinates are **extraneous information** that the typical user probably won't need to accomplish their task of selecting a section of the screen they can visually see. Furthermore, the cursor turning into a "+" alone does very little to instruct the user how to make the selection and does not warn the user that once they click and drag their cursor, they will not get a chance to confirm that selection or make any modifications prior to the section being captured in a screenshot. The relationship therefore is not made explicit between the "+" cursor and the results of moving it.

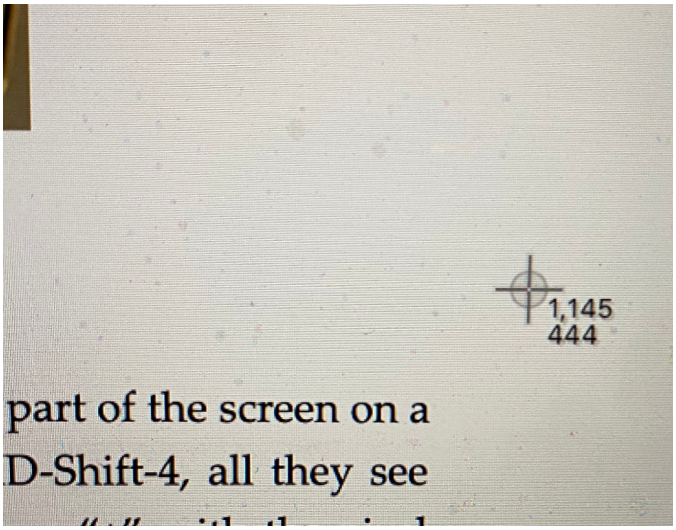

Figure 3—Mac laptop screenshot cursor.