Assignment 1, HCI Principles, Gulfs, Feedback Cycles

Manick Mahalingam (mmahalingam6@gatech.edu)

Question 1. Processor vs Predictor Model, A Hybrid Approach

Processor model in HCI, which is based on Behaviorism focuses largely on *objective, outcomes, optimization of a task performed by the user.* Canvas platform for reading Courses module used by various students across the globe concentrates on getting a user to speed up on navigation of screens, understand the layout of various components to work on a particular course, build muscle memory with motor responses. This helps the user follow a mechanical guideline rather than a cognitive approach, makes it a core foundation for accomplishing the tasks of reading books, listening to lectures, participating in forums, work on assignment rather not focus on improvising actual interface, number of clicks to reach to a page, external interface like Ed Discussion. Processor models helps to establish a platform for any interface to play with the fundamental aspect of the tasks within expertise limit, rollout to wider spectrum of users at the predictor model. Any existing students who have been in OMSCS for a while, could be examples of Processor model user where they become expertise on above concepts

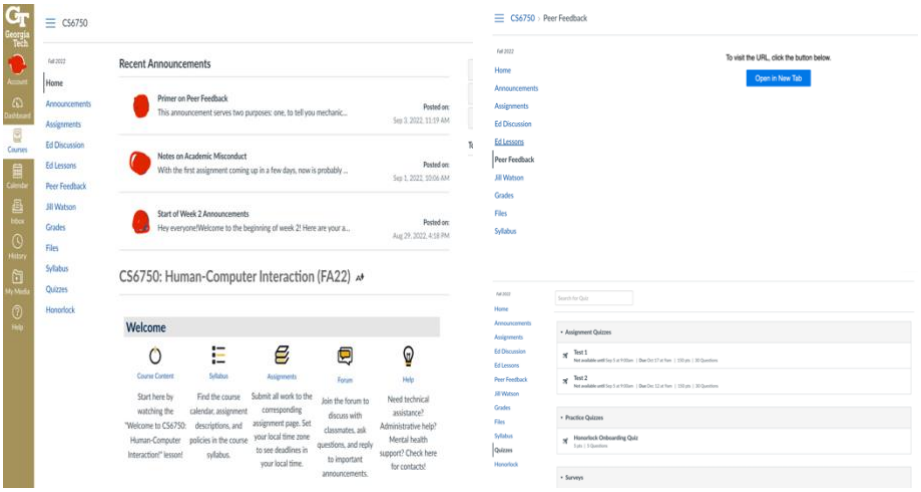


Figure1.1 *-- Layout of Canvas showing references to external link and outdated design*

Predictor Model approach in HCI, tries to extract information from various user perspectives on not only improvising UI experience but also focus on the outcome

of certain tasks, example, Courses getting expanded to be used in mobile devices with a new OMSCS app for easier listening, integrating Ed Discussion, Lessons, Peer Feedback within the same application. Piloting some of these features for experts to toggle between legacy and new interface, conduct pop-up reviews, polls or surveys within the application window, opinions from new users by providing scaffolding prototypes, track the usability with metrics to find out the best possible outcomes and bring them in practice

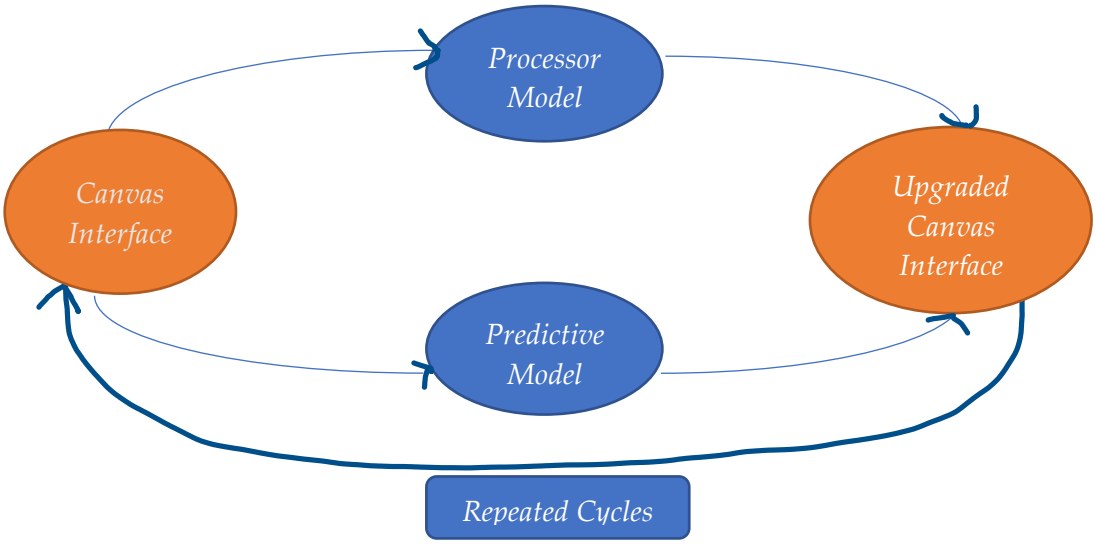


***Figure1.2** -- Proposed design to improve Canvas Interface using hybrid approach*

For generations, perspective of human over computer interaction changes or improvises to next level, to keep with new era of interfaces in an AI future world, reading courses, working on assignment, communication, should become more interactive, so *hybrid approach of combining Processor and Predictive model in cycles* would help improve Canvas. While the processor focuses on inputs received through predictive not steer away the objective and efficiency of tasks performed in Canvas through design optimization, integrate a training model, predictive could help in cognitive analysis of the outcomes for the same task by experimenting and extracting information, collect data points, providing prototypes. Example, Canvas courses has some of the components within, whereas lessons, discussion, feedbacks are external, this could be integrated together and UI/UX design can be enriched with new features and release pilot versions or toggling, get onto feedback cycles to review, implement the design and

strategy with traceability, compare existing usage by analysis and follow HCI's fundamental principles would make Canvas, a leading cutting-edge UI/UX design

Question 2, Participant Model, Youtube Music

Youtube Music is one of my favorite applications that is used throughout the day in laptops, mobile device, smart watches, speaker systems, automobile console, voice control and more. The interface of it is something that could fit in as a best candidate for participant model view of a user and to *emphasize the experience in various contexts*. Personally, I use it at work, running, driving, relaxing, household chores

While at work, sitting at a desk, human cognitive resource focus actively on the application as various features can be explored including suggestions from previous playlist, listening to new recommendations, choosing from various genres, artists and albums, download albums for offline listening, most importantly application can be trained with its featured ML algorithms to provide suggestion when not actively navigating or using the application. However, passive listening can be integrated better with voice controls for search, pause/play, next song, and active listening interface features can be improved with lyrics for songs can be prompted by holding on to an album, finding additional information of a song while hoovering on an artist, settings on behavior, sound quality, time limits, sharing album suggestion to friends

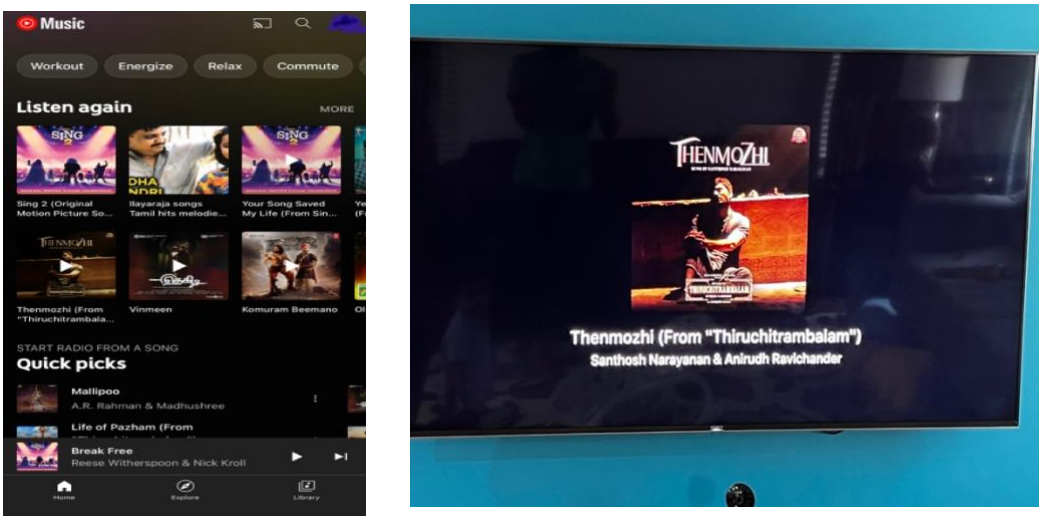


Figure 2.1 *–Youtube music, Participant Model example Interface at work and home theater systems*

While running or workout or running chores, the primary focus is to complete the activity whereas interface acts passively for a user to perform the task of listening to songs, here voice controls add less value compared to smart watch or mobile device navigation of the application, so notification or locked interface in mobiles/watches provides basic metadata and navigation, this could be challenging as we could lose sight of a favorite album over suggested random music. To overcome this challenge, voice controls featured on smart watches or headphones, or mobile device could be used to control the interface in choosing an album or song desired by a user, another option is to hoover on smart watches or locked mobile phones to choose albums

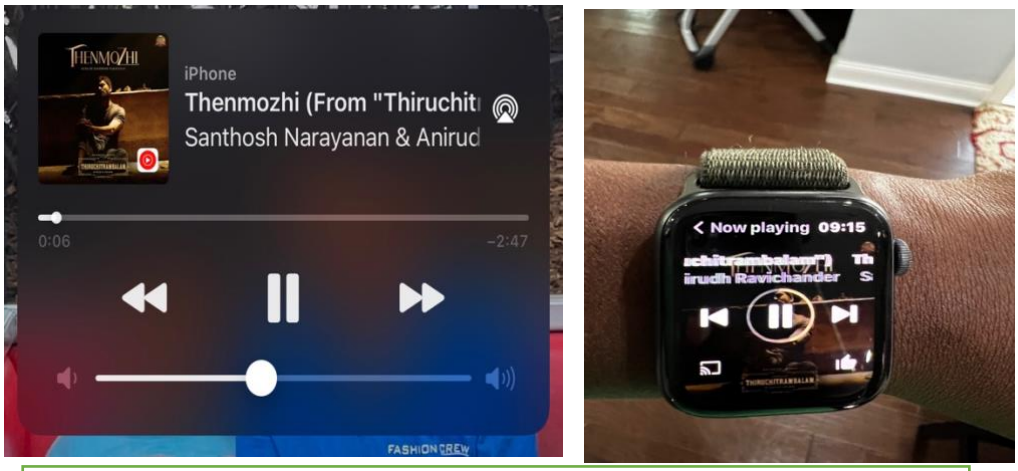


Figure 2.2 -- *Participant Model Interface, shows constraints of outdoor running and driving*

Both the above contexts get mixed up while driving and improvising those challenges could be of great use in a driving context, user can explore the features on scenario of sitting at a desk while stopping at red lights or stuck in traffic or waiting for assistance in breakdown (hope that doesn't happen) whereas explore scenario on running would equate to driving which should limited or no focus on the application but driving. Voice controls could help immensely in this use case

Another context that could be discussed to *explore the interface while using with speaker systems*, home theaters, partying, grand stage events where the detailing of sounds from the music plays major factor and versatility of genres in music can be experienced with different sound. Recommendations, searching suggestion, voice controls can be maneuvered whereas playing around on sounds is a challenge or externalized or mechanical, this challenge can be resolved by integrating the

complex music patterns into the interface where experts could explore various sound patterns, play multiple genres of songs in various culture under one roof, bring karaoke version to provide user and environment participate into the real feel of enjoying music

Question 3, Ed Discussion

Ed Discussion is a forum or portal to interact with Professor, TA, staff and fellow students on various categories of questions within a subject or more. As a novice user to Ed Discussion, the student's intention is to get answer(s) to their question(s), the *Gulf of Execution's Identify Intentions* stage, where student knows the goal or task to perform. Now to perform this task, user plays around the interface to *identify actions* to post a thread by choosing a course, click on 'New Thread', a title, a category and post. Additionally, student gets resources on existing threads, various categories, notifications, replies on threads, people online and more, but these are helpful for any future tasks or actions that can be identified to make a student get used to the system and became an expert with a cognitive thought process that goes into the memory even gets into the subconscious memory, but the *execution in interface* as identified, goes through the same sequence as defined earlier and wait for a response

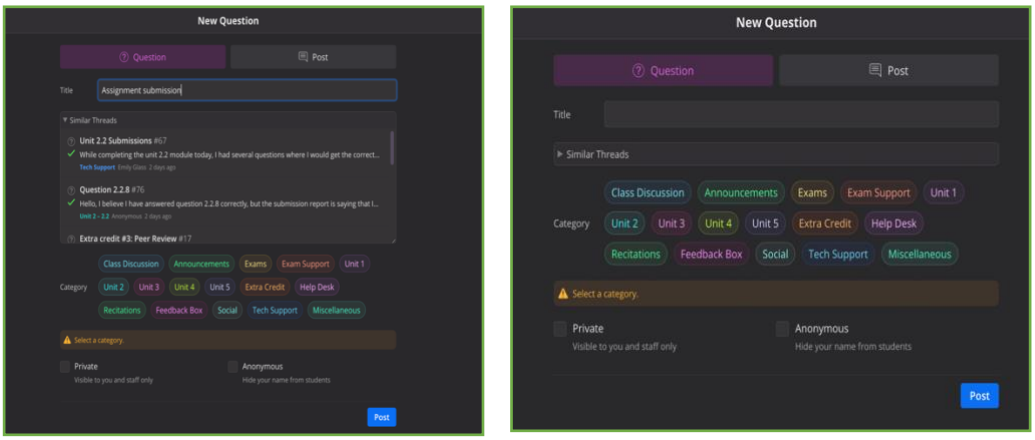


Figure 3.1 – Identify Actions to post a question in Ed Discussion

Now there is an additional scenario that could be encountered by the student, where they identify another action for a similar task, Figure 3.1 shows similar threads toggle that could show related threads for their question, which could slightly deviate from the previous execution of interface, but the response time

could be quicker or instant, student get answers for the threads post by peer students immediately and could end up satisfied on the task

Once the student completes his cycle on execution, the feedback that is received is something the student might be novice about, so that brings to bridge (Gulf) of Execution, where the output received in the interface is to show their thread with a blue dot, at the top of threads list, student interprets by going through the output, evaluates to realize that his task is not completed, or they have not received an answer for the question.

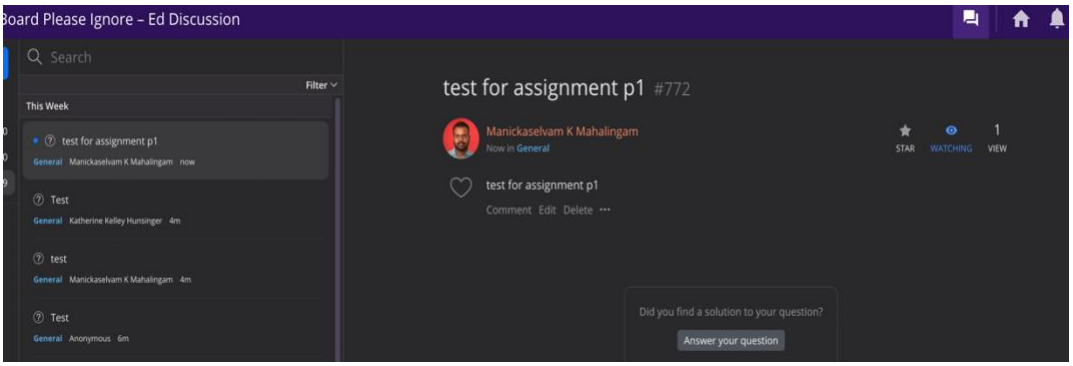


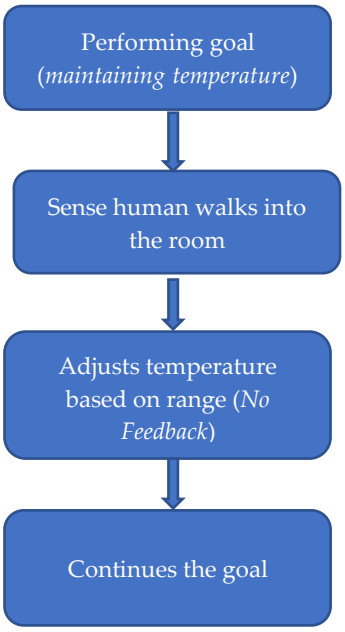
Figure 3.2 -- *Perceive, Interpret Feedback received in Ed Discussion Interface*

Though the outputs are received in the interface in different dimensions, includes showing up in the list of threads and notifications with bell icon, there is no way the student knows when or how they would receive an answer (*large gulf of evaluation*), but assuming that once an output is received from other participants they are perceived by the student, process them in the mind, interprets the answer and evaluates it by reading through the thread to be convinced that it's an answer to this question

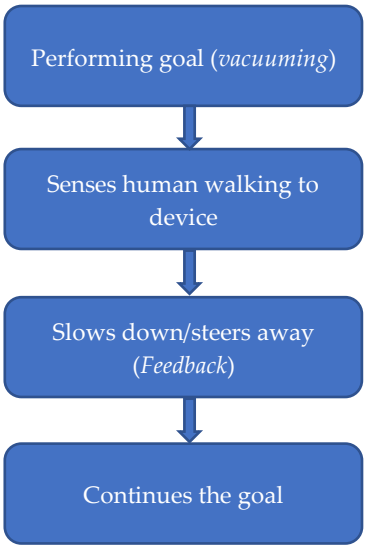
Now the additional way of receiving an output is notification by email which is out of the interface, follow-ups on the same thread goes throw the same cycle but narrowed gulf of execution and evaluation. "Similar threads" could also provide instant feedback where the user could go throw other suggested thread and evaluates the goal

Question 4, Sensors in Roborock vs Ecobee

Ecobee is a smart thermostat that helps to set temperatures on for different schedules, turn on heat/cool, range of temperatures (say 70-76), alert for change of filters, senses smart away and hold additional sensors to communicate in individual rooms to dynamically adjust its temperature based on whether room is occupied or unoccupied. When the room is occupied, it falls in temperature range, if not, goes above 76 or below 70 depends on seasonal climate. The goal here when a person enters the room, it should sense a human obstacle with its additional sensor, turn to occupied, adjust the temperature within the range and the person enters room should receive feedback with the feel of temperature while its obvious it would take few minutes, does the person know if it sensed their entry? here the *gulf is wider* as interface takes few minutes to update to "occupied" so as the temperature and there are no instant feedbacks that the room is occupied even with additional sensors



Roborock is a smart vacuum system that helps to automatically clean the house, this device has a smartphone interface, voice controls, voice notifications, Lidar sensors, stronger design to setup schedules, add maps for various floor, zone cleaning and more. The goal here is when any of its sensor senses a change in its map, for example, it creates a floor map during the first run using it sensors, (*narrows bridge of execution and evaluation*) and constantly looking for feedback to update its schedule, now when running into the device while it is performing the goal, it senses a human obstacle with the legs and steers away from there dynamically without any hassle. This is by for a great example of *narrow gulf* where the feedback is instant, the sensor identifies a human intervention, immediately responds with feedback of steering away from them



Lesson learnt from Roborock, the gulf is narrower due to the fact on training its Lidar Sensor with multiple feedback cycles of execution and evaluation, applying

a hybrid approach of processor and predictive model, one example of a feedback cycle on the context of this scenario would be placing human mannequin in a floor (*plan*), sensor to identify interruption (*specify*), continues to run the device (*perform*), in response, device steers away (*perceive*), understands the sensor activity (*interpret*) and device did not stop (*evaluates*), this could be done in repeated cycle until it reaches to a saturation of narrowed gulf.

Figure 4.1 – Ecobee Thermostat Sensors (Wider Gulf) vs Roborock Lidar Sensors (Narrow Gulf) Mickey is a human here

The same interface could be applied to Ecobee's room Sensor, an example of a feedback cycle here would be to place sensor in a room, human enters the room (*plan*), sensor to identify motion (*specify*), runs HVAC to change temperature, notify with a voice "adjusting temperature, please wait" or a constant beep or notify in smart app, or main thermostat console (*perform*), in response, person feels HVAC run with a noise, gets notified over voice or application interface or beep (*perceives*), interpreting of HVAC run or beep could be difficult compared to notifying over voice or application interface (*interprets*) and compares the notifications and range in temperature, for sensor functionality of sensing motion (*evaluates*), this improvement done in repeated cycles could help narrow down the bridge on thermostat and provide a better user experience

References

- canvas.gatech.edu
- https://edstem.org/
- music.youtube.com
- https://www.ecobee.com/en-us/
- https://us.roborock.com/