2020

ACCESSIBILITY ENHANCED BY SOUND AND HAPTICS

LOYOLA UNIVERSITY CHICAGO

ANDREW LITTLETON – MS COMPUTER SCIENCE

Table of Contents

Abstract

The Accessibility Enhanced by Sound and Haptics (AESH) project was designed to analyze and refactor accessibility features within home gaming consoles using enhanced audio and haptic feedback to benefit those requiring the features. Initial tests gauged reactions and ease of use of the narrator by the test subject within the three major game consoles; PlayStation 4, Xbox One, and Nintendo Switch. Ease of access features with greater flexibility and user experience customization will broaden the number of users with disabilities that may want to use the system but have been discouraged from doing so by either societal isolation or simple inability to functionally use the system. This project primarily develops for and focuses on user experiences for those that are hard-of-sight. The Xbox One held the most robust set of accessibility features and was chosen for further testing.

The Xbox One features the most robust accessibility features relative to the other two consoles. The PlayStation 4 includes accessibility features limited to only common and frequently accessed windows. The Nintendo Switch does not include any accessibility features. While all three consoles perform their intended task of being a platform for digital interactive entertainment incredibly well, inclusion of those with greater sensory needs is of varying priority to each company. Ease of access features are intended to provide greater amounts feedback to the user from the system in ways that cater to the user's needs. The AESH project uses audio modifications and the introduction of haptic response through the controller to provide the user with alternatives to the already present accessibility feedback options. This project works to develop features and modifications exclusively for a game console's core operating system and it should be known that this project does not intend to modify the game development process to include the ease of access related features discussed. The audio and haptic assistance developed for this project are intended to be used in low stimulus system environments and gameplay within video games does not provide a suitable environment for this particular set of accessibility features.

Research for haptic and audio response development was based on the ideas of psychological conditioning intended to invoke unconscious or semi-conscious responses to stimuli from the console. Audio development was based on core sound design techniques and audience response understanding. This relied on the idea of "bright" affirmative sounds and "dark" negative sounds. Haptic feedback research was based on video game control schemes and discussion about how users effectively interact with the controller to perform actions as well as receive feedback. Existing audio was enhanced to make the Xbox One narrator more concise.

Narrator adjustments and haptic features were considered beneficial and audio considered beneficial but unnecessary. Each feature was important but would be beneficial to add for accessibility features on operating systems as they will give users that need it an extra level of customization and eliminate some feelings of discouragement related to the stigma of using ease of access-based features.

Research

Accessibility Research Study (ARS)

Initial research was done on the already existing ease of access features for each console through a user study. The test subject involved was hard-of-sight and unfamiliar with the navigation practices relating to each console. The same subject performed each trial of the study. The study was setup as a series of trials divided between consoles. All trials began from the home page or initial landing screen for each console and the user was instructed to perform a specific navigational task. "Actions" were left to the user's discretion. The goal of this study was to gather an understanding of the existing accessibility features strengths and limitations while also gauging user experience.

<u>Trial 1: Xbox One</u>

- Task: Navigate to Microsoft Store
    - Result: **Succeeded**
    - Comments:
        - Narrator's voice gave user excess feeling of stress, was too verbose and should cease speaking when cursor moves
        - Announcement of sorting order for structures would be helpful
        - Haptic feedback would be helpful to indicate cursor position behind the narrator's voice
- Task: Open a Game/Application and perform action within it
    - Result: **Partial Success (Not platform related)**
        - User opened app but was unable to perform action
    - Comments:
        - Microsoft does not mandate accessibility features be turned on along with Xbox's, so the application opened, and user was unable to perform any intentional action

<u>Trial 2: PlayStation 4</u>

- Task: Open My Profile and perform an action
    - Result: **Failed (Platform related)**
        - Accessibility feature announced use was not supported in the My Profile window so no action could be performed
    - Comments:
        - Narrator voice is far less stressful to listen to relative to Xbox and was more concise
        - Features were unavailable in areas of interface that were common for user interaction, but not mandatory
- Task: Open a game and perform an action
    - Result: **Partial Success (Not Platform related)**
        - Test subject launched game but could not perform action as the game did not include accessibility features.
    - Comments
        - The PS4 game that launched did not include and accessibility features so upon launch, the user could not perform any intentional action

<u>Trail 3: Nintendo Switch</u>

- No trial performed as it was discovered the Switch does not included any accessibility features

The results of the trials were heavily favored towards the Xbox One. The Xbox included the most detailed and developed ease of access features, including:

- High Contrast Cursor Outline
- Narrator
    - Multiple voice options
- Magnifier
- Closed Captioning
- Controller button remapping

These features allow for a more customizable ease of access environment for those with differing accessibility needs. The PlayStation 4 included some, but not all of the above features. It included a narrator and a magnifier. The narrator was mentioned by the test subject to have a more pleasant voice than any of the Microsoft ones, but was considerably lacking in detail. Though this did not pose a problem to the test subject. The Nintendo Switch only included one ease of access feature: the magnifier. Though this is technically considered an accessibility feature, it was determined to be completely unhelpful by the test subject.

Xbox One

Given the research results, it was determined that the Xbox One would be the primary development and research console. The Xbox One's operating system's primary user interface is developed in a similar fashion to Windows 8 and 10. It uses a tile-based menu with floating side bars as necessary. The system structures different pages as a list of collected tiles belonging to each page. Each tile has name associated with it and possibly a description. During testing it was noted how the narrator reads information present on the screen. It reads in the following order:

1. Announce the page
2. Announce the structure of tiles the cursor is placed in
3. Announce the. Number of options available in that structure
4. Announce any extraneous options (repeats on movement to different structure until screen changes)

On some menus, this method of description might not be effective when compared visually, but when heard exclusively as audio, it is beneficial to listen to. Knowing this, development on making the narrator more concise involved removing the repetition of the fourth step and tightening pauses.

The Xbox One's ease of access features, while already robust, still left room for criticisms. The greatest criticism was against the narrator. The narrator lacked the ability to stop talking when the user moved the cursor off of a tile and was also repeating a variety of unnecessary information.

Audio and Haptic Development Research

For users that are hard-of-sight, audio and tactile based feedback are the primary forms of accommodation. This can be seen with portable audio devices providing descriptions in museums, braille, a white cane, etc. These tools and practices were developed to allow those that need them to operate as close to a societal normal as possible.

Sounds for AESH were developed using classical and operant conditioning methods. A bright sound is played as positive reinforcement to indicate a reward from the system or a dark sound as positive punishment to indicate the input/action was invalid.

The existing audio enhancements modified for this project were:

- Narrator (Microsoft David as default voice)
  - Concise and Verbose versions
- Cursor Navigation Response
  - Affirmative and Negative responses

The haptic feedback was designed and developed with the console's controller in mind and would not require additional hardware for the user. The Xbox One Controller (*Figure 1*) was the device used in testing.

Figure 1: Xbox One Controller with Motor Position Marked

Operant and Classical Conditioning for Feature Design/Development

The AESH project's aim to give console user's more robust ease of access features does require some learning and responding from the user. Designing effective HCI interfaces relies heavily on certain unconscious assumptions and responses from anyone that interacts with a system or application (Hayward, 2020). To help the user learn to effectively use and make assumptions about the console's interface, operant and classical conditioning were chosen as the most effective method of unconscious user training. Feature development was focused on aiding the user through certain tasks but also training them to respond to audible or haptic stimuli in the appropriate manner.

Pavlovian classical conditioning relies on the basic framework of pairing stimuli with a time and space is a significant enough environment to produce a conditioned response (Gottlieb & Begej, 2014). This concept is ultimately based on the role of taking a stimulus and applying an already existing response to a subject's response to it. The idea of Pavlovian conditioning has been expanded upon to explain that incremental trial-based processes relying on training time, the number of sessions, and the number of trials to effectively reinforce the strength of association between the stimuli and the conditioned response for the subject.

Though operant and classical conditioning are similar in nature, the difference lies in their focus. Operant conditioning relies on the idea that frequency of a certain behavior is governed by its consequences (Murphy & Lupfer, 2014). AESH development and testing was done using single subject behavioral observance. Single subject testing utilizes a baseline condition, very little change between sessions, and introduction of a treatment condition until stability is reached. Operant conditioning requires a reinforcer to govern behavioral modifications in the subject through either reinforcement or punishment. Reinforcement and punishment, in this case, are defined as the addition of a positive stimulus or the addition of a negative stimulus, respectively.

Applying these two conditioning methods to AESH would require an understanding of the user's knowledge of the console interface. If the user was new to the console, then number of training time and number of sessions became more important to focus on for the classical conditioning tactics. If the user already had experience using the console, then number of sessions became more critical to reinforce learned responses. AESH did not seek to ingratiate new responses in the subject, but to simply reapply existing ones. The baseline condition provided to our subject during the research experiment was the simple idea to complete the task. Each task was ultimately the same: "Use the provided accessibility feature and any knowledge [The subject] have of the console's interface to guide [The subject] to the intended location." The subject had little knowledge of the system interface itself outside of the affirmative (select) and the negative (back) buttons and their respective functions. The initial experiment was meant to establish the initial operating conditions for further testing and to decide on the most effective treatment condition to produce stability. The selected treatment conditions would be modified audio feedback, the introduction of haptic feedback, or a combination of the two. User stress level is the variable used to check for stability. Stability would be considered reached when the subject feels the accessibility features no longer provide stress and have assisted them in their user experience.

Haptic Perception and Feedback

Haptic perception in a regular environment relies on tactile stimulation through mechanical events, such as a person feeling the temperature of an object as they touch. Though this action may seem discrete to adults, a child could consciously decide to touch an object with the explicit purpose of determining its temperature. This example is meant to illustrate the foundations of haptic response in common everyday life. Haptic feedback in a virtual environment is very similar. The major differences lie in the nature of what provides the tactile simulation and the types of mechanical signals generated. Virtual environment haptic feedback requires a haptic interface to produce the mechanical signal rather than the subject engaging with the object. Through this, the mechanical interaction is simulated to produce the effective tactile stimulation. Haptic interfaces can simulate most natural mechanical signals, but also has the distinct advantage of being able to produce non-natural ones as well.

Haptic interfaces require certain hardware and software connections and interactions to effectively produce the simulated mechanical signals desired. The interface often requires the following devices at minimum (Robles-De-La-Torre, 2008):

- Actuators: The physical means of providing the simulated mechanical signal.
- Sensors: The device's means of measuring the current physical state of the device and the actuator operation data.
- Power/Electronic Control Software: The control of all the electrical requirements for the device.
- Software Toolkit: The primary programmable controller for the device. Used to establish more detailed operation and programmatic settings for the interface.

The Xbox One game controller features two motors as its primary actuators. It uses the two motors to create simple vibration feedback for video games. The left motor provides low frequency vibration and the right motor provides high frequency vibrations. Both are fastened to the controller's chassis to allow for the different motor vibration patterns to be effectively combined as game designers create different haptic feedback experiences. Though the motors are fastened to the chassis and intended to be used in tandem, both units can still be felt from their individual positions so long as the other actuator is inactive.

When developing for haptic interfaces, perceiver expectations are important to consider (Robles-De-La-Torre, 2008). For the Xbox One controller, it must perform for video games as a constant source of feedback for the player. It serves as a means of communicating game interactions more expansively through the player's primary source of those interactions. Though the controller plays an active role as an interface in a video game, if it were to also serve as an active interface during moments that are non-game related then the design and style of the haptic interactions must be diametric to one another. If the feedback from the controller is intense during stress inducing events in a video game, then the feedback must be as calming as possible as a player cycles through menus to maintain the diametric nature of the design. Maintain a design of opposites helps to teach the subject which vibrations patterns belong to which stress level and to which events.

Development and Testing

Development and testing were done in the same fashion as the initial ARS. The experiment retained the single subject and simple goal format but removed the other consoles. The subject was given a location within the Xbox's navigation system (Store, an app, etc.) to locate using the modified assistance features. Due to the nature of the Xbox operating system and Microsoft's proprietary development criteria, testing had to be done without a true connection between the integrated elements and the Xbox One. The goal of testing was to determine which user experience modifications lowered or increased stress in the subject, using the default test as the control reference.

Development for each of the audio modifications and the haptic features were based in lowering the stress of the user. Considerations for the user's experience with technology and the Xbox One itself were taken into consideration. The narrator's audible response was tuned to adjust for user experience as well as listening ability. For instance, a user with experience using technology but unfamiliar with the Xbox One's navigation and menu design might be comfortable with the default and could transition to a more concise version. Though a user with no Xbox One experience that is skittish around new technology might require a more descriptive and slower narrator to ensure the user learns and understands what is on the screen and what they are able to do. The haptic feedback was designed with the intention of relieving user stress from being unable to adequately see the screen for any reason. It was meant to serve as an alternative to the cursor highlighting on the screen. The haptic feedback was designed to be most effective with the narrator enabled but can also be used without it.

Haptic Experience Design and Testing

To adequately develop the haptic feature set for the project, the system level operation of the haptic interfaces had to be understood. Microsoft provides the XInput API with their distributions of development tools for Win32 applications. The API provides insight on how the Xbox controller communicates with applications. These applications are often video games and the documentation reflects this by often using the term "frames" referring to graphical rendering frames used to perform visually reference real time calculations for players, though the applications using XInput are not required to follow this design and development scheme. The Xbox One controllers use a simple state machine to govern vibration. The function XInputSetState function is explicitly designed to manager the vibration states of the controller.

The function is passed a vibration structure among its parameters to effectively establish the vibration effects.

The motor vibration has a force magnitude output range equivalent to that of an unsigned integer (0-65,535). This allows for 65.5 thousand force magnitude steps to establish greater flexibility in vibration strength and experience. The force magnitude delta between frames in video games is not be high enough for the vibration experience to be jarring as the curve between any given vibration values can still be felt without it disrupting the user experience. Taking this concept to menu interactions, consideration for user focus had to be taken. Players in games are focusing on many different things at once so vibration effects must be strong enough to draw attention but not so intense as to distract them. In a menu, the player/user is not as engrossed in what is happening on the screen so vibration values can be more distinct without as much mechanical work. Developing the haptic user experience is ultimately done through experimentation and user reviews. Initial values for subject testing are provided in the following table:

Table 1: Motor Speeds and Cursor Action (Speed values are % of 65,535)

Cursor Action	Active Motor	L Motor Speed	R Motor Speed	Duration (seconds)
Up	Both	25%	50%	1
Down	Both	25%	50%	0.5
Left	Left	50%	0%	0.5
Right	Right	0%	75%	0.5
Select	Both	10%	20%	0.25
Invalid Op	Both	25%	50%	1.5

The values for each action and motor were selected to give actions greater variety in vibration experience though some are still similar to one another. The “Up” and “Down” actions are intended have similar haptic responses because they are an intuitive directional pair, but the “Down” function is given the shorter time value because down is a more common direction to move when using the Xbox navigation layout. Invalid operation was given a longer duration because it is intended to clearly discourage the user from performing that action. Introducing the ***invalidCursorOperation()*** haptic feedback will provide the necessary positive punishment required to train the user against performing invalid operations while the other functions will serve as necessary mechanical stimuli so the user will be able to determine cursor actions without being able to see it.

The simulation was programmed using functions for each cursor movement called by a more robust ***vibrate()*** function. The ***vibrate()*** function is provided the length of time the controller will remain in the vibration state with motor speeds set by each cursor action function.

```
XINPUT_VIBRATION vibration; //vibration structure
DWORD controllerState; //controller state object
Xbox1HapticSimulation simulationObject; //simulation object to govern whole vibration process
int vibrationDuration; //length of vibration in seconds
```

```
void Xbox1HapticSimulation::vibrate() {

	//conditional to get keystroke and call coordinating function
	if (keystroke.Flags == XINPUT_KEYSTROKE_KEYDOWN) {
		switch (keystroke.VirtualKey) {

		case VK_PAD_LTHUMB_LEFT:
			duration = leftCursorMove();

		case VK_PAD_LTHUMB_RIGHT:
			duration =  rightCursorMove();

		case VK_PAD_LTHUMB_UP:
			duration =  upCursorMove();

		case VK_PAD_LTHUMB_DOWN:
			duration = downCursorMove();

		case VK_PAD_A:
			duration = selectCursorAction();
		}
	}
. . .
	//set duration = return value of coordinating function
	controllerState = XInputSetState(0, &vibration); //indicate controller vibration
	std::this_thread::sleep_for(std::chrono::milliseconds(duration)); //wait for designated
duration period in ms (set by coordinating cursor function)
	simulationObject.stopVibrate(); //stop controller vibration

};
```

The duration passed to the function by each cursor action's return value is the used to tell the application to wait for that period of time before calling the ***stopVibrate()*** function which zeroes all motor speeds and sets the controller state accordingly. Motor speed values are explicitly determined for the sake of consistency within the experiment. The following code snippet demonstrates how motor speeds and vibration states are established:

```
//sets left motor speed to 10% and right motor speed to 20% and  duration to 0.25 seconds
int Xbox1HapticSimulation::selectCursorAction() {
	vibrationDuration = 250;
	vibration.wLeftMotorSpeed = 6554;
	vibration.wRightMotorSpeed = 13107;
	return vibrationDuration;
};
```

Accessibility Narrator and Audio Modifications

Developing the narrator modifications was directly influenced by the feedback from the Accessibility Research Study done during the research phase of the project. The most notable feedback came from the subject mentioning the lack of conciseness and repetitiveness from the included narrator. Using this feedback, two new narrator modifications were developed. The first is a concise version that strips the bloat phrases and repetitiveness from the narrator's audio as well as lower the amount of time between phrases and the second is a more verbose version that elaborates in greater detail what is on the screen and potential common actions.

When audio is played within a virtual environment, digital signal processing and audio memory management are critical to achieving desired performance without dropping samples. When the narrator speaks, it does not play one long audio file but stacks many on top of each other for different phrases. Assuming the Xbox One's native audio uses audio files with a 48 kHz sample rate with a bit depth of 24, the amount of time each narrator phrase takes to be load and unloaded to and from memory is relatively small compared to the amount of time used between phrases (Spanias, Painter, & Atti, 2007). Keeping this in mind, shortening the narrator's phrases and cutting the amount of time between them required more care processing considerations. Though the considerations were necessary, the Xbox One features hardware specifications that could adequately process audio in significantly shorter amounts of time.

To shorten the narrator's speech, many of the phrases were either shortened to eliminate unnecessary statements, such a description, and repetitive statements were only played once. The more verbose design built on the initial default design but slowed the narrator down for those that may need time to process the information as well as including greater details regarding what potential actions the user could perform. The potential are phrased as, "Move [Direction] to go to [Potential Location]". Demonstration of the modified narrator was read by the tester to the subject. The following chart describes the narrator timings according to action for all three narrator modifications. The action performed was to go from the Home to the My Games and Apps screen.

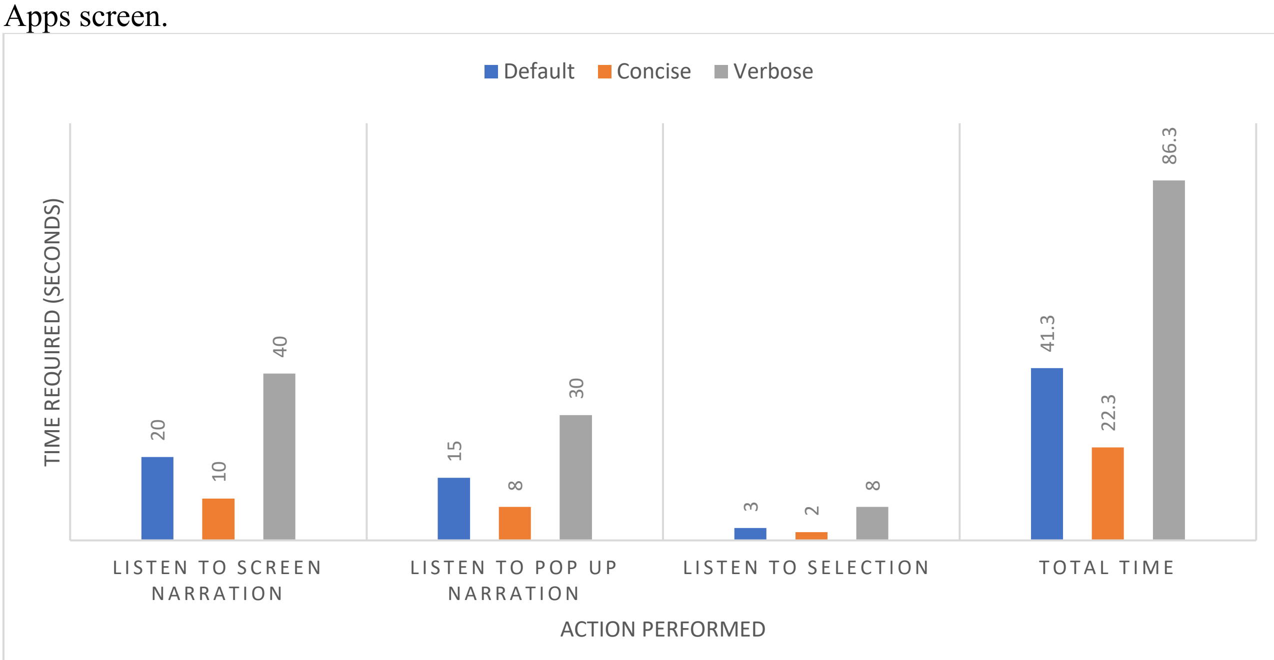

Chart 1: Graph of Narrator timings based on modification series

Each modification was timed against the default. The chart does not include representation for button presses or repeated action timings, but their times were summed and are present in the, “Total Time” series.

Different cursor sounds were added as well to provide audible reinforcement or discouragement feedback. The newly introduced positive punishment sound was designed using the existing affirmative section sound. It was pitched down and given more presence with the 400 Hz – 2kHz range. This was to ensure the already bright default sound was established as the positive reinforcement and the negative sound, due to being sonically diametric to its counterpart, would serve as the positive punishment sound. Desktop operating systems were the inspiration for the design and implementation of the new audio sound effects. Desktop systems include a variety of warning and notification sounds. The new audio feedback was designed, not to indicate a fatal problem, but to be a warning similar to when a user tries to perform an action outside of a mandatory action window.

Post Development Testing

The single subject test was divided by treatment condition introduction and always began from the home screen. After each trial, the subject was told to rate their stress level on a scale of 1 to 10 during the experience. The subject was also told to listen to the narrator phrases in their entirety without attempting to skip them. It proceeded as follows:

<u>Trail 1: Concise Narrator</u>

- Task: Navigate to Microsoft store
- Result: **Succeeded**
- Subject Stress Rating: 4
- Comments:
  - Concise narrator is significantly easier to listen to
  - Subject did not feel bloated with information
  - Subject did express concern for missing/forgetting information as previously repeated statements were no longer repeated and a distraction caused the subject to miss an announcement.

<u>Trial 2: Verbose Narrator</u>

- Task: Navigate to any application and launch it
- Result: **Succeeded**
- Subject Stress Rating: 2
- Comments:
  - Verbose narrator was thought to be helpful in the case of someone requiring it, but not for this particular subject
    - Subject belongs to age group statistically noted to pick up technological interface nuances more quickly than others
    - Initial ASR and previous trial gave subject enough insight into Xbox One navigational schemes
    - Subject felt bored listening to narrator and wished to advance the task

<u>Trial 3: Haptic Experience and Sound Effects</u>

- Task: Navigate to an entertainment app through the “Entertainment” group
- Result: **Failed**
- Subject Stress Rating: 6

- Comments:
  - Subject failed to complete task because the haptic experience provided no assistance to the subject’s ability needs
  - Experience would be beneficial with greater experience and a navigational aid
  - Positive reinforcement and punishment techniques were effective

Trial 4: Haptic Experience/Sound Effects and Concise Narrator

- Task: Navigate to the Achievements Page
- Result: **Succeeded**
- Subject Stress Rating: 1
- Comments:
  - Subject felt the addition of the narrator made the navigation and user experience significantly easier and less stressful
  - Haptic experience did not feel comfortable, but the intention was present and understood

Trial 5: Haptic Experience/Sound Effects and Verbose Narrator

- Task: Navigate to the Achievements Leaderboard
- Result: **Succeeded**
- Subject Stress Rating: 2
- Comments:
  - Similar comments to previous trial
  - Verbose narrator was helpful to listen to, but this particular subject did not require navigation details in this way (see Trail 2 comments).

Results and Conclusion

Through testing, it was understood that additional audio responses, though beneficial were deemed unnecessary by the test subject. Haptic feedback was considered extremely helpful in acknowledging cursor movement but was at its greatest effectiveness when paired with the narrator. Lowered stress in subject as haptic feedback gave illusion of participation rather than simple “lecture and consider” concept. Despite this, the motor variations and vibration patterns would need to be reconfigured to provide a better experience. The vibrations were identified as too intense or simply “strange”. The vertical direction feedback shook controller more violently than subject’s preference and jarred them from the intended experience. The “Select” action response felt strange. It was noted to be effective, but uncomfortable to the subject. Alternating side motor distinctions remained slightly too similar despite different speed percentages. This is likely due to the motor’s connection to the controller chassis. After further inquiry, the subject elaborated that the motors could be felt from their respective positions but shook the body of the controller too significantly for the desired effect to be achieved. The haptic experience was deemed helpful and effective at reducing subject stress only when provided with an audible navigational aid and ineffective otherwise.

The feedback provided about the modified narrator was consistently positive. The concise narrator proved to be helpful and stress reducing in the event that the subject has previous experience with the Xbox One system. Though experience and user comfort are subjective, it can be said that if the user is comfortable navigating the Xbox One without much thought, it can be assumed the concise narrator would be the narrator design they would use if they needed it. The verbose narrator was noted to be an effective ease of access feature for those unfamiliar with Xbox One navigation or those that have trouble remembering how to effectively navigate it. Its lengthier descriptions and potential action mentions were considered to be a welcome addition. The newly added audible feedback is not as critical of an addition as it did not stand out enough against the narrator to be as effective. When paired exclusively with the Haptic experience, it achieved the desired psychological conditioning effect, but not the desired user experience.

The additional features presented would allow the Xbox One to enhance its accessibility features to those that strive for the closest to normal configuration of the system. Narrator adjustments and haptic features were considered beneficial and audio considered beneficial but unnecessary. Each feature was important but would be beneficial to add for accessibility features on operating systems as they will give users that need it an extra level of customization and eliminate some feelings of discouragement related to the stigma of using ease of access-based features. These adjustments to the Xbox One's ease of access features will help those with more varying accessibility needs and will help them better tailor their user experience to something they would prefer rather than be forced to use the limited options available. Part of the AESH project was to build from the existing accessibility features on the Xbox One. Building from the initial and already proprietary technology Microsoft has available, development and implantation of these features will not require a significant amount of labor or time. Broadening the nature of the Xbox One's ease of access features will benefit Microsoft as a company looking to maintain inclusion within their gaming market by allowing users with more unique accessibility needs to feel comfortable using the system. More robust ease of access features will benefit all of those involved.

Works Cited

Dotsenko, A. (2017). *Designing Game Controls*. Retrieved from Gamasutra: https://www.gamasutra.com/blogs/AndrewDotsenko/20170329/294676/Designing_Game_Controls.php

Gottlieb, D. A., & Begej, E. L. (2014). Principles of Pavlovian Conditioning. In F. K. McSweeney, & E. S. Murphy, *The Wiley Blackwell Handbook of Operant and Classical Conditioning* (pp. 5-11).

Hall, D. E. (2002). Structure in Music. In D. E. Hall, *Musical Acoustics* (pp. 435-450).

Hayward, N. (2020). COMP 341 - Human Computer Interaction. Illinois, United States of America: Loyola University Chicago Department of Computer Science.

Lumen. (2020). *Classical and Operant Conditioning*. Retrieved from Lumen Learning: https://courses.lumenlearning.com/atd-fscj-generalpsychology/chapter/classical-and-operant-conditioning/

Murphy, E. S., & Lupfer, G. J. (2014). Basic Principles of Operant Conditioning. In F. K. McSweeney, & E. S. Murphy, *The Wiley Blackwell Handbook of Operant and Classical Conditioning* (pp. 167-180).

Robles-De-La-Torre, G. (2008). Principles of haptic perception in virtual environments. In I. G. (Ed.), *Human Haptic Perception* (pp. 363-379). Birkhauser Verlag.

Spanias, A., Painter, T., & Atti, V. (2007). Audio Signal Processing and Coding. John Wiley & Sons.