Computing Tools for the Psychological Study of Choice Behavior

A Senior Honors Thesis

Submitted in Partial Fulfillment of the Requirements

for Graduation in the Honors College

By

Cheryl Limer

Computer Science and Psychology Major

The College at Brockport: State University of New York

May 11, 2018

Thesis Director: Daniel Rogers, Lecturer, Department of Computing Sciences

Table of Contents

Abstract

The computing tools described here were developed with the help of Daniel Rogers for Dr. Lori-Ann Forzano's psychology lab at The College at Brockport, and were designed to aid in the study of choice behaviors, such as self-control and impulsivity, across age ranges. Impulsivity can be defined as the choice of a smaller, less delayed reward over a larger, more delayed reward, with self-control as the opposite. These two concepts are often studied in psychological contexts, and computer programs such as these allow for the study of self-control in regards to video reinforcers. The current programs recreate and expand upon a previous program used in the lab in order to assist in future studies. During development, close work was done with psychology faculty in an effort to examine the underlying psychological principles to be studied with the completed programs. The design and development was done in a way that allows the programs to meet the standards necessary for them to be used in scientific research. These programs are compatible with newer technologies, and improve upon previous versions of the program by allowing retrieval of videos from additional sources, allowing new studies to be developed and run.

Computing Tools for the Psychological Study of Choice Behavior

Introduction

Psychological Background

Impulsivity is often defined as the choice of a smaller reward, or reinforcer, after a shorter delay as opposed to its opposite, self-control, which is defined as a choice of a larger reinforcer after a longer delay (Forzano, Michels, Sorama, Etopio, & English, 2014). These opposing phenomenon have been linked to life outcomes such as substance abuse, diet and general well-being, among others, and are therefore often studied in various situations and with various rewards, or reinforcers (Goodwin, Browne, Hing, & Russell, 2017). As self-control and impulsivity are aspects of daily life and contribute to various life outcomes, it is vital researchers have an understanding of who is more likely to be more impulsive or have more self-control and in what scenarios each is more likely. Logue, King, Chavarro, and Volpe (1990) found that impulsivity is impacted by changes in amount of reinforcer and delay length, while Forzano and Logue (1994) found that impulsivity can also be affected by the point in time during the session during which the reinforcer is received (i.e., either during or after the session).

Additional studies have expanded upon these findings and have begun to look at how the type of reinforcer might impact self-control (Forzano et al., 2014). There can be many differences between qualities of different reinforcers. For example, there is the difference between primary reinforcers such as food, which can be consumed immediately, and secondary reinforcers such as money, which are used as an intermediary step to get a different reinforcer (Forzano & Logue, 1994). Odum and Rainaud (2003) demonstrated that this difference in reinforcer can impact impulsivity, finding different levels of impulsivity for the primary reinforcers, food and alcohol, than for the secondary reinforcer, money, with generally more impulsivity when presented with a primary reinforcer rather than with a secondary reinforcer. Often, self-control is studied in relation to consumable reinforcers, such as food or drink, in order to examine specifically primary reinforcers. Even within the category of primary reinforcers, however, different qualities that affect self-control can be found. Studies have therefore began to look at the differences in proportions of self-control between different primary or consumable reinforcers.

Generally, when using a primary reinforcer, food or drink is the reinforcer used, however some studies have used other primary reinforcers such as online entertainment, and have compared the self-control proportions of participants exposed to varying reinforcers. Some examples of online entertainment that have been used in self-control studies are video games, used by Millar and Navarick (1984) and more recently video cartoons, used by Navarick (1998), among other studies. While online entertainment is not consumed in the same way food and drink is consumed, it is still considered a primary and consumable reinforcer due to the fact that it, in and of itself, is the reward, and it is not a reinforcer that holds value in relation to an additional reinforcer, as in the example of receiving money. When money is used as a reinforcer, it is not the money itself that is a reward, but what the money could then be used on, whereas when food, drink, or online entertainment is used as a reinforcer, it is directly rewarding (Millar & Navarick, 1984).

It is valuable to compare participants' impulsivity in relation to differing primary reinforcers due to the fact that primary reinforcers can differ in qualities that may impact self-control. For example, Forzano et al. (2014) compared self-control between two primary reinforcers, juice and video, that varied on multiple dimensions such as metabolic value, satiability, and level of habituation, all of which may impact impulsivity (Charlton & Fantino, 2008; McSweeney, 2004). Specifically, it is possible for the participant to become habituated to the reinforcer and satiated throughout the course of the study. This has the potential to then change their preference for the reinforcer, which could impact the results of the study (Forzano & Logue, 1995; Forzano, Porter, & Mitchell, 1997). Additional consumable reinforcers must then be compared with food or drink to see whether or not the satiability of the reinforcer plays a role in their tendency to choose impulsively. In order to compare two different consumable reinforcers, the reinforcers must be delivered in the same way to ensure that any difference in impulsivity is a result of the reinforcer and not the way it is delivered. Because of this, a computer program was developed to deliver the video reinforcer the same way juice had been delivered in previous studies (Forzano & Shunk, 2008).

Original Application

In 2008, a program was written for use in Dr. Lori-Ann Forzano's psychology lab at The College at Brockport: State University of New York. This program was written in C#, and was designed to mimic the Skinner Box used in previous studies to deliver juice. This physical apparatus has a rod at the center that the participant can either push to the left or right, with one side being associated with a shorter delay and a shorter reinforcement period, the impulsive choice, and the other side being associated with a longer delay and a longer reinforcement period, the self-control choice (Forzano & Logue, 1992). When using this apparatus, the reinforcer is juice. Therefore, when the rod is pushed to one side, there is a measured delay time and then a straw is available for the participant to drink juice from for the reinforcement period. Once the reinforcement period ends, the straw is taken away and the participant is cued to choose again. These cues to the participant come from a series of lights on the apparatus. There is a white light at the top of the apparatus. When it, and the disks on either side of the apparatus, are lit, the participant is cued to move the rod and make a choice. Once a choice is made, the white light is extinguished and a delay light is lit. There are two delay lights, one for each possible choice. On the left there is a green light that is lit when a left choice is made. On the right there is a red light that is lit when a right choice is made. The appropriate light stays lit for the entirety of the delay period and the reinforcement period. The delay light is then extinguished and the middle white light is relit once the reinforcement period ends, and the discs are recolored when the participant can make another choice. Each choice, along with its delay and reinforcement period, is considered one trial; a series of trials where the delays and amounts are the same for every trial make up one session. A participant attends 3-6 sessions, where the delay and amount times vary between sessions, for the completion of the study, depending on which study they are participating in (Forzano et al., 2014).

In order to compare results obtained using this apparatus with results obtained with video as the reinforcer, the mechanism used to deliver the video must mimic as closely as possible the apparatus used to deliver the juice. Therefore, the Skinner Box program written in 2008 was designed after the physical apparatus, as seen in Figure 1. During each trial, the participant sat in front of a computer screen on which there were two buttons for selections and three discs meant to mimic the lights. Similar to the physical apparatus, the middle white disc was lit to cue the participants to make a choice, and once a choice was made by clicking a button, the corresponding delay disc was lit. Once the delay period had ended, the video selected for the session was played for the entirety of the reinforcement period. The program then returned to the screen with the buttons and discs for the participant to make their next choice. The setup of the application, including picking the video and setting the delay and reinforcement period lengths, as well as the results obtained from the session, were contained within this program which would draw from and write to other folders on the computer (Forzano & Shunk, 2008).

This program was used in Dr. Forzano's lab for various studies from its completion in 2008 until the Spring semester of 2017. It was at this time that all of the computers in Dr. Forzano's lab were updated to Windows 10, and the program no longer ran on the computers. In order to continue to run participants in Dr. Forzano's lab, it was imperative that the program be updated or rewritten to be compatible with the Windows 10 operating system.

Method

Initial Work With Original Source Code

The source code for the program was obtained from the original developer and examined. It was found to use a C# package, DirectX, which had since been deprecated. Past its deprecation it had been compatible with later versions of Windows even if not fully supported, however it became incompatible with Windows 10. Research was conducted to determine if there was an appropriate substitute for this package that would perform the same functions that the DirectX package had. Two possible solutions were found, however neither one was able to mimic DirectX closely enough to be inserted without significant code changes. At this point it was decided that it would be more beneficial to completely rewrite the program using newer technology that might be compatible with machines for longer period of time rather than update an otherwise out of date program.

Program Re-creation

Due to familiarity and availability of resources, it was decided that the new application would be written in Java. JavaFX was then chosen to create the user interface due to its ability to contain and play video files. It was also decided that minor usability changes would be made with the re-creation pertaining to the setup of the program, in addition to a new function that would allow two sessions to be run back to back, therefore minimizing the number of times a participant would need to come into the lab. This function, when employed, will run two sessions in a row with the delay and reinforcer amounts switched, so that for the second session run, the self-control and delay choices will be on opposite sides from the first session. Prior to this function, the participant would have to come into the lab for each session. By putting two sessions back to back, the program will cut the number of sessions the participant has to come in for in half, which will likely increase the percentage of participants to complete all of the sessions of the study.

Throughout the development of the re-creation it was important to work closely with Dr. Forzano and her research assistants who will be using the program in the lab to ensure that it functions as closely as possible to the physical Skinner Box and is easy to learn and use. In order to create a program suitable to Dr. Forzano and her lab, the development process had particular focuses.

Precision. In order to be considered a legitimate, precise and useful tool in a scientific arena, programs used in psychological experiments must meet certain standards. This specific program requires precise timing of features of the user interface and video reinforcement. The first attempt to control this timing was made using the Thread.sleep command which pauses execution for a specified amount of time. While this paused execution on the current thread, it did not pause execution on the thread that the video was on, causing the code to not have any observable effect during execution. After attempts to control the separate threads, it was determined that this would not be a workable solution. After some exploration, a more workable solution was found and Timer objects were used. These objects allow for a pause in time without the creation of new threads. When nested, one timer controls the delay from when a choice is made until the video begins playing while the inner timer controls the delay while the video plays until the end of the reinforcement period. A third timer is then used to pause the program from the end of the video for the rest of the trial time until the next trial begins. These timers measure the passage of time in milliseconds and are therefore as precise at controlling the delays as possible.

This program must also be precise in terms of its similarity to the physical Skinner Box used to deliver juice as well as to the original program. As seen in Figure 2, the user interface of the current program closely replicates the user interface of the original program, which was a replication of the physical apparatus. In addition, there are certain small details in the execution of the trials that need to be replicated in the program. For example, the circular discs on the screen that function as the lights, as well as the buttons the participants select, must change color at certain times in response to a participant selection or the passage of a specific amount of time. The logic behind these color changes of the various aspects of the user interface are all included within the program. In addition, in order to provide the participant selection feedback similar to what they would receive pushing a physical bar from side to side, a click sound is included in the current program when a button is clicked. This confirms for the participant that they made a selection and that the response was acknowledged, which is necessary to avoid confusion and also to as accurately as possible recreate the response feedback from the physical apparatus.

The program also requires precision in its collection and display of results. In addition to recording which option is selected for each trial, a series of other information regarding each trial must be recorded and added to the results. The program requires a record of the time it takes a participant to make a choice, referred to as the latency, to be output as a result. Therefore, the time from when the trial starts, or when the participant had the ability to make the choice, until a choice is made is measured in milliseconds and added to the results from each trial. In addition, the number of times they attempt to click a selection button during a delay period, when selections cannot be made, is also recorded per button per trial and added to the results document. The results document itself is laid out in a way that is clear and concise, clearly differentiating the results of each trial from each other trial, listing the participant information and the length of the delay and reinforcer periods, and calculating the percent self-control without rounding. Having a clear results documents created by the program during execution helps ensure accuracy in terms of the program documenting results as well as maintaining accuracy by eliminating mistakes or misreading when the results are transferred by hand from the results document to a data analysis program.

The program also attends to precision in regards to checking input for possible errors, and trying to minimize human error when information in being input. This is done on the setup screen by ensuring information has been entered into every field before the user can continue to the next screen. The program also checks that all fields, with the exceptions of participant information fields, are filled when saving the entered information as a preset to be used later, while also checking the name of the preset to ensure it is a unique name and will not overwrite other saved information. On the video selection screen, the video can be tested before it is selected by being played on the selections screen. This test ensures that the video selected is the correct video and also allows for a check of volume before the session begins. The program also makes sure that the test video playback has stopped before the video can be selected. If two videos need to be selected for the session, the first video selected is not included in the options for the second video so that the same video cannot be selected twice. These checks of user input allow for increased precision during execution by catching or not allowing certain human errors.

Flexibility. This application is intended for use in a psychology lab in which new studies and procedures are constantly being introduced and put into practice. For that reason, one focus of the development was to create an application flexible enough to be used in multiple ways for multiple studies, some of which have yet to be designed. One way in which that is done is through the ability of the program to deliver multiple reinforcers. The main intention of the application is to deliver video as a reinforcer, however it is also equipped to deliver points as a reinforcer. Using the application in this way, following the delay after the participant makes a selection, a blue button appears which the participant can click as many times as they want for the entirety of the reinforcer period in order to earn points. These points can then be exchanged for a reward of some kind at the end of the session or study. By including this functionality, this program can be used in additional ways for a multitude of different studies.

Similar to the original version of the program, this program was developed with the intention of being able to be used by all ages, adding flexibility in order to allow studies to be designed for participants of all age groups. This was done by ensuring simplicity for the participant during the session. While setting up the program, the instructions for the participant can be changed, allowing for different phrasing depending on the age of the participant, or even no written instructions if the program is being used with children who cannot read. In addition, any video saved as an .mp4 can be used as a reinforcer, allowing for different video reinforcers targeted at different age groups. The use of the program itself has no necessary reading, complex thought, or complex motor movement on the part of the participant, making it possible for a participant of any age to learn and use.

Ease of use. This program also required design focused on ease of use for the researchers setting it up for participants to use. Due to the high turnover rate of the student population working in the lab, there is always a new researcher learning all elements of the lab and of running participants. Because there is so much to learn at any given time, it is important that this program require very few instructions for an experimenter to learn in order to use it correctly. This ease of use was partially achieved by limiting the number of screens involved for setup and execution while also clearly labeling all text fields. The previous version of the program required the experimenter to execute a few steps, ill-defined by the user interface, to browse through the file explorer at two separate times: once to pick the information to be loaded into the program, and once to choose the video to be played during the session. It also had three screens for setup, not including the additional screens that were used during the browsing process. In addition, in order to run the program, the user had to go to the ‘Simulation’ option in the taskbar and select ‘Launch Program’ which was not labeled clearly enough to find without instruction.

As demonstrated in Figure 3, this version of the application simplifies that process. The information input on the third screen of the original application regarding the participant has been added to the first screen of this application. In addition, as opposed to opening tabs at the top of the application to get to the file explorer to then find the correct folder and document, this version of the application has a dropdown of possible options included in the original setup page, as well as a dropdown to select the correct video on the next screen. These options are .txt documents that contain all of the information required to run a particular session. The dropdown is labeled as ‘Presets’ and allows the experimenter to pick the associated .txt document for the session they are running, and clicking ‘Apply’ autofills all of the information on the setup screen. This feature’s ease of use is also vital to maintain precision, as it makes sure that the information entered is the same every time a given .txt file is used, minimizing the risk of typing mistakes on the part of the experimenter when information essential to the session is being entered. In addition, executing the program can be done by selecting a button labeled ‘Continue’ which is easier and more self-explanatory than launching the program in the earlier version.

This preset function also assists in the flexibility of the program due to its ease of use, allowing new preset documents to be created from the setup screen. Information pertaining to a new study or new session can be entered on the setup screen and saved as a .txt document to use at later times by clicking a ‘Save Entered Values as Preset’ button at the bottom of the screen. Presets can be deleted by deleting the .txt document it is based on. The process for this is outlined for experimenters in a protocol regarding the setup and use of this application, as seen in Appendix A. This ease of use pertaining to the creation and deletion of preset values leads to increased flexibility in creating and running additional studies with new, different values.

Additional Program Development – YouTube Application

Rational. Upon completion of the Skinner Box program re-creation, a discussion began as to how a major update to the program could add more flexibility and allow for new studies. It was decided that adding the capability to play videos from YouTube, in addition to the selection of videos downloaded and converted on the computer that have been used in previous studies, could be beneficial for multiple reasons. Including videos from YouTube allows for new studies to be developed that could not be run before. Studies can be done regarding specific types of videos, for example music videos versus plot based videos versus pop culture videos, etc. Expanding the number and types of videos that can be used as a reinforcer allows for the study of intricate details of the reinforcer that may impact self-control.

In addition, allowing participants to pick videos from YouTube ensures that the participant finds the reinforcing video desirable, thereby increasing the validity of the results obtained. With the previous program, participants chose their desired reinforcer video from a list of cartoons which had been downloaded, converted, and used for a number of years. The cartoons chosen were popular at the time and purposely chosen to appeal to both children and adults who may participate in a study using this program. When choosing from a list, however, it is possible that a participant does not really like any option or may choose arbitrarily, making the reinforcer not appealing and therefore less effective. In addition, TV shows become outdated quickly, and it was found that the children participating in studies recently were not familiar with any of the cartoon options, again making the reinforcer less effective. By allowing each participant to choose their favorite video from YouTube, it can be assumed that each participant finds their reinforcer desirable and appropriately reinforcing while also eliminating the possibility of videos becoming out of date.

Original design and obstacles. The original design for this additional functionality was to have the participants choose a YouTube video at the time of participation that would then be streamed from YouTube during the session in real time. It was thought that the setup for the session would be the same with the exception of the selection of the type of reinforcer. If YouTube was selected, instead of having the experimenter select a video from the dropdown menu of available videos, the participant would be given a screen in which they could enter a search term and choose from the first few results from YouTube with that search term. The video ID would then be held, and the program would go to the URL of that video and play the video for the appropriate length of time during reinforcement periods. Although this was able to be achieved in a prototyped version, this solution was deemed ineffective due to a lack of control over various aspects of the video.

Streaming directly from YouTube did not give the program enough control over when the video would play. The URL had to be accessed each time the video was to be played, creating a load time at the beginning of each reinforcer period. Because the loading time depends on internet speeds at any given time as well as traffic on the website itself, this lag was not consistent and therefore could not be accounted for or standardized across trials, sessions, and participants. In addition, each time the URL was accessed, the timestamp included in it needed to be updated so that the video started from wherever it had stopped at the end of the last reinforcement period. This time was increased by adding the reinforcement time to the previous time stamp, however the lag affected the amount of time the video actually played for, making any algorithmic calculation of the timestamp inaccurate.

This solution also did not control for the advertisements played at the beginning of some videos on YouTube. Once again, there was no algorithmic way to determine which videos had advertisements and how long those advertisements were, should the program play them out of the participants view in some way. In addition, sometimes the advertisements even played when starting a video in the middle and did not just play at the beginning of the video. It was vital that the participants not have an advertisement as their video reinforcer, and so the lack of control over this aspect of streaming videos straight from YouTube was not optimal.

Finally, streaming videos from YouTube at time of participation caused some videos to become not embeddable, as certain copyrighted videos are not allowed to be embedded in any way. It is important for the study that the video is embedded; the other parts of a YouTube screen would be distracting for a participant, and would allow them to focus on or go to a page other than the chosen video. The study requires that participants watch only the video, with nothing else on the screen, throughout the reinforcement period. Due to the embedding requirement, certain copyrighted videos could not be embedded and therefore could not be chosen. This had the potential to severely limit the videos that could be chosen by the participant, and was not an optimal solution.

Redesign. Upon determining the original solution to be unworkable, it was decided that instead of having participants choose and stream the video at the time of participation, they would choose the videos from YouTube ahead of time. An experimenter would download these videos to the correct format and location so that they could be used in the same way the other videos are. Downloading the chosen videos before the participant arrives completely eliminates the issue of advertisements, gives the required control over the loading, timing, and playback of the video, and ensures all videos can be viewed embedded with no other options for the participant but to watch the video. This solution required an additional application, as opposed to a change to the Skinner Box application.

Copyright considerations. In order to use this solution, copyright laws were examined in order to verify that the videos could be used in this way. Any video that is not copyrighted can be downloaded and watched by anyone without violating any laws, therefore this issue only pertains to videos selected by the participant that are copyrighted. This use of videos is within the limits of Fair Use policy, found in Title 17 U.S. Code § 107 of copyright law. This policy, cited in its entirety in Appendix B, states that “the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as…research, is not an infringement of copyright” (Copyright Law of the United States, 1998). As these copyrighted YouTube videos are being used for educational purposes with no commercial gain of any kind, and because the clips of copyrighted works that are being used are short enough, the use of copyrighted works through downloading from YouTube is covered under this section. An additional portion of this policy states that a factor to be considered is “the effect of the use upon the potential market for or value of the copyrighted work” (Copyright Law of the United States, 1998). In order to adhere to this factor, it was decided that the videos will be deleted once they are used for one session. This limits the impact on the number of views determined by YouTube, as it will not be viewed when it is converted but will be viewed exactly one time as a copy, maintaining an accurate view count which may impact the value of the work.

Design process. This application was not based on any previous application or process, and therefore was completely designed and executed within the scope of this completed thesis. The basic design of this application involves two distinct functions. The first function allows an experimenter to enter information about a particular participant and the study being run, such as participant number, age group of participant, minimum length of video required to have enough video for all reinforcer periods in a session, and number of videos needed to complete the study. This information is then used to create an identifying code that is given to that participant. The application then sends the code and a link to the application in a predetermined email. The other function of the application is from a participant end, where the participant enters the code they are given and is taken to an embedded YouTube page where they browse for and select the videos they would like to use in the study. The selections, if they meet certain requirements, are sent to the psychology lab’s email address, where an experimenter can download and convert the videos to have them ready for the session.

As this application was developed, there were a series of design and implementation decisions that were made in an effort to make the application easy to use and to minimize the possibility of errors on the part of the experimenter or participant. Therefore, a major design focus was to make the process as automated as possible, thereby minimizing the amount of steps an experimenter or participant has to remember and execute while also minimizing the places in which a human error could be made. To do this, as many of the processes as possible are controlled by the application, such as the creation of the code and the email sent to the participant. The code creation requires information from the experimenter, input through a user interface seen in Figure 4, however the code itself is created in the application in a specific format so that the information encoded in it can be obtained by the application or the experimenter at a later time. The code was created in the following format, where the description in parenthesis describes the information the parenthesis would be replaced with: ‘P’(participant number)(‘A’ if the participant will be choosing the videos for themselves or ‘C’ if the participant is a child with a parent choosing videos for them)(5 random characters)’T’(minimum minute length necessary for a video)’M’(minimum second length necessary for a video)’S’(number of videos to be chosen). Because the code is generated by the application and sent in a predetermined email with the link to the application without the experimenter seeing it, both the link and the code are known to be correct when they reach the participant. Similarly, the application sends the links to the videos that the participant selects directly to the experimenter. In both of these cases, the application handles certain steps so that copying errors on the part of the experimenter in regards to the code, link, or email and on the part of the participant in regards to the video links can be eliminated.

The application was also designed to automate aspects of the video selection process, including how many videos are to be selected and which videos fit the criteria necessary to be selected, minimizing errors in this arena as well. The graphical user interface, as seen in Figure 5 and viewed by the participant, is created dynamically, depending on information contained in the participant code. The right hand side of the application contains a series of buttons used to select the videos for the study. As different studies contain different numbers of sessions, it is possible that the number of videos necessary would vary from study to study. This information is added to the participant code so that the application, when creating the user interface for the participant, can dynamically add and create the necessary number of buttons that is exactly the same as the number of videos to be selected. This ensures that neither too many nor too few videos are selected and that the video selection process only has to happen once per participant for the entire study.

Similarly, the participant code holds information regarding the criteria the videos selected have to adhere to. The videos selected must be longer than the minimum length found in the code, must all be different from each other, and must be appropriate for viewing in a psychological study. The application is able to check if the selected video meets all of these criteria before the video is sent to the experimenter to be converted. By not allowing participants to select videos not meeting the criteria, the application ensures that all of the videos are usable before they are selected and sent to the experimenter. This eliminates a scenario in which the participant chooses a video not meeting the criteria and the experimenter checks and has to ask the participant to repeat the video selection process. It also ensures that all videos are able to be used in the study at the time of participation, eliminating a scenario where a video is erroneously deemed to meet the criteria but then is found not to during the session, invalidating the results of that session.

Implementation considerations. Implementation of the design previously discussed requires interactions with entities outside of the application itself. The application interfaces with YouTube in order to obtain information about the selected video and interfaces with Gmail in order to send emails.

Interfacing with YouTube. Interfacing with YouTube through the application allows the application to analyze the information about the video to ensure it meets the necessary criteria to be selected. This is done jointly through the functions provided by JavaFX and through the YouTube API provided by Google. The YouTube page embedded in the application is controlled through a JavaFX class in which a change in the URL of the embedded page can be retrieved. However, because YouTube uses AJAX to load links within their webpage, the page is only loaded in part when the URL is changed, and the JavaFX object could not detect the change in the URL. Therefore, when a video is to be selected, the button clicked refreshes the page, loading it all the way, so the URL of that video with the video ID can be retrieved. The video ID is then parsed out, compared against the other videos to make sure it has not already been selected, and saved.

The remainder of the interaction with YouTube happens through Google's API. In order to have access to this API and the associated .jar files, registration with Google is required. A key is then obtained that must be used in the code before any Google functions can be run. This application uses two Google functions, each of which gets some information about a video given the video ID. Once the video ID is obtained, it is sent to these functions to make sure the video is longer than the minimum necessary length and to make sure the content is appropriate. YouTube has certain content filtering processes in place, one of which makes the user sign in and prove they are at least 18 years of age before viewing certain videos. This filtering decision is recorded and can be obtained through one of the functions provided by Google. Therefore, if YouTube filters the video so that the viewer must be of a certain age, it cannot be chosen for any study in the lab.

In the case that these APIs change in a way causing them not to work in the future, error checking was put in place so that while the videos will no longer be checked based on length or content, the application can still be used.

Interfacing with Gmail. The application also required interactions with an email account, which was achieved using JavaMail. JavaMail requires a username and password of a Gmail account with limited security settings, allowing the application to log into it and send emails from it. Originally, the design of the application had the emails being sent from the email address of the lab. However, upon further consideration, it was determined that this was not an optimal solution. Because the application is to be used on many different computers and sent between computers, the account information has to be hardcoded into the application. As the lab email will be used often by many people, hardcoding the login information into the application was a risk, considering the login information for that account might have to be changed at some point. If that account’s information changed, the application would no longer be able to send emails, which is a vital part of the application. Therefore, instead of using the lab email account to send the emails within the application, a dummy email account was created to send the emails from. This Gmail account is not used by anyone except the application, therefore it can be more safely assumed that the login information will never have to change, meaning that the application will always be able to send emails from this account.

Final functionality. In its final form, this application is saved on computers in the lab and has also been converted to an .exe file which is uploaded to the lab's Google Drive. When a new participant is scheduled to participate in the study, an experimenter uses the application to enter the information about the participant and study and sends the email to the participant with the generated code and the sharable link to the application that is uploaded on Google Drive. At this time, the participant follows the link to the application, downloads it, and uses it by entering their code on the first screen. They then browse for and select the number of videos, meeting the criteria, necessary for their study. The experimenter waits for the participant to download and use the application on their own time and checks the lab email account for the email from the application with the videos selected from the participant. Once this email is received, the experimenter opens the email and uses an online YouTube to .mp4 converter to download and convert the selected videos, naming them appropriately and moving them to the Video folder associated with the Skinner Box program. The protocol written describing the use of this application to experimenters working in the lab can be found in Appendix C.

Results/Discussion

Recommendations for the Future

There are certain elements to both of the discussed programs that are not optimal, either due to configuration limitations or necessary byproducts of certain design decisions. These elements have workable solutions, however they could be improved in the future.

Skinner Box 2 limitations. Due to design decisions, the Skinner Box 2 program needs to be setup in a specific way with specific associated files in order to run. The .jar file used to execute the program must be in the same location as three folders with specific names: Presets, Videos, and Results. These folders contain information the program either uses (Presets, Videos) or creates (Results). The program is written so that it looks in folders with these names in the same location of the program itself to find the files it is looking for. Therefore the file path can be dynamically found, but the folder name is hardcoded. This was to organize the necessary files in categories and also to allow the program give the user the file option directly in the user interface. In the previous version of this program, files used could be anywhere on the computer, but during the setup users had to open up a file browsing window and know where in the file directory those files were. By having these files in specific locations, the application can find and present the file options to the user during setup without requiring a browser window and without requiring the user to know where the files are located. While this makes for a more difficult setup of the application itself due to the specificity of folder names and file paths, it does make the setup process for each session easier while using the application, which, of the two processes, is the one that is done more often.

In order to make this suboptimal setup usable, a document has been created with detailed setup instructions to be used whenever the application has to be placed on a new computer. This document, which also contains information regarding how to use other aspects of this application, including the maintenance process for presets which has already been discussed, can be found in its entirety in Appendix A. In addition, all of the necessary folders, as well as the application itself, are currently in a clearly labeled folder, so that folder, in its entirety, can be copied on to a new computer and will have all the necessary components to run the application. In the future, this application could be modified to search in multiple folder locations for files that could be potentially used in the execution of the program. If this application was modified to dynamically find appropriate files, the limitations due to setup could be eliminated.

In addition to requiring specific folder names in the same location as the executable file, the JavaFX class used to play the videos requires a file path with no spaces in any of the file names. This means the video name itself, as well as the names of all of the folders to get there, must have names with no spaces in them. While this was not optimal, the class used was able to do everything else that was needed, and therefore was chosen even with this limitation. This issue is mitigated by having detailed instructions regarding how to name the videos and suggesting in the setup instructions that the main folder holding all of the Skinner Box 2 materials be placed directly on the C: drive to keep the file path short, thereby minimizing the folders needing names without spaces. In the future, this program could possibly be updated to use different resources to control video playback that do not require such a specific file path.

YouTube application limitations. The YouTube application also has certain limitations due to both design decisions and resource limitations that impact its use. The application has a series of related issues that are a byproduct of the design decision to have the application encapsulate as many processes as possible in order to minimize potential human errors. By adding as much automation to the application as possible, the application becomes less flexible in certain ways. For example, by having the application send the email with the code and the link to the application to the participant, this email has to be the same and cannot change without a source code change. While this source code change would not be particularly difficult, it does require someone with a programming background. A reminder email containing the exact same code also cannot be sent to a participant, as the experimenter does not see the generated code until the videos have been selected. In addition, the application loses the same flexibility in terms of the instructions displayed during the video selection process. Should those instructions need to be changed for a given experiment, it would also require a source code change. Finally, if the application is to be used in the lab at time of participation rather than ahead of time, the video selection process still sends an email, adding an extra unnecessary step given the situation.

Given these limitations that were a known byproduct of the design choice, it was still determined to be more beneficial to try to minimize human error than to make this application particularly flexible, as flexibility for this program was not quite as important as it was for the Skinner Box 2 program. In order to try to account for these limitations once the design had been decided, the instructions in the email and within the application were written in a general way that would not likely need to be changed soon. In addition, a few variations of the email and instructions were accounted for, with one variation for adult participants and another for child participants. The instructions in both the email and in the application were also created to be slightly dynamic. In each, there is a sentence informing the participant of the minimum length of video necessary for selection. The sentence is changed based on the minimum length denoted in the generated code so that the instructions and email always accurately and directly tell the participant how long the video needs to be. In addition, developer documentation was left so that these changed could easily be done by a developer. This developer documentation can be found in Appendix D.

An additional and more influential limitation of the YouTube application is its reliance on Java to run. The participant, in order to choose videos, must download and use the application, however it can only be run on Windows computers that have Java installed. Upon investigation, it was found that less people have access to computers that fit this criteria than originally thought. It is also not ideal that the application has to be downloaded at all, as computers do not all react in the same way to downloading an .exe file. Therefore, it was hard to give the participants detailed instructions on what they have to do in order to download and run the application. There is also a safety concern with downloading unknown .exe files, and while it is known that the application the participants are getting a link to does not contain a virus, it takes a certain amount of trust on the part of the participant to download an unknown application to their computer. In addition, some computers have processes in place to prevent this security risk. Embedding the application in the browser so that it could be run through a URL was considered but not deemed to be a valid solution because, while there are resources to do that, they are all depreciated due to security risks and browsers are removing support for the plug-ins needed for the URL to work. Therefore, it would be a solution that would soon be unusable and would still require the participant to configure their browser in a specific way.

The current working solution to this problem is threefold. The first part is based on the fact that all campus computers are Windows computers with an updated version of Java installed, and all college students can be assumed to have access to a campus computer. The email to be sent to college student participants was updated to reflect that if their personal computer was not a Windows computer with Java installed, they should use a campus computer to download and run the application. This also eliminated some variation in the download process as most campus computers will process the download in the same way. This is, however, not considered a good long term solution.

Furthermore, this solution was not considered a viable solution for child participants as their parents who are using the application cannot be assumed to have access to a campus computer. Therefore, a different solution was created for use with child participants. When the participants come in for the first session, after the consent form is signed, the parents will select the videos and the experimenter will convert them, and then the child can use the program with the selected video reinforcers. In order for this solution to work, an additional feature was added to the YouTube application on the experimenter end. When entering the information to generate the code, instead of sending an email to the participant with the code, the code itself can just be generated and displayed. The experimenter can then copy and paste the code into a new window of the application and have the video selection screen ready when the participant arrives. This is also not considered a good long term solution as it requires extra time during the first session and wastes time while the child's parent is picking videos and also while the experimenter is converting and downloading the videos. In addition, as the application was not intended to be used in this way, it is a little cumbersome to generate the code and get the video selection screen ready for the participant, and the video selections are sent via email to the experimenter, adding the extra step of getting the videos from an email instead of directly from the participant.

These two parts of the solution are workable, but not optimal for long term use. The third portion of the solution, therefore, is for future versions of this program to be written purely as a web application and not as a Java application so that it can be run directly in a browser by giving the participant a URL. This solution would not require any specific computer configurations and would therefore eliminate all of the current problems associated with downloading and using an .exe file. This potential project will be pitched to computer science students at The College at Brockport in the Fall 2018 semester. It is recommended that a computer science student transfer the participant side of the video selection application to a completely web based format based on the current version of the Java application and with input from Dr. Forzano. This would allow video selection to be done in a web browser and eliminate reliance on Java which would then allow any potential participant to select videos on any computer prior to coming into the lab.

Future Uses

These two programs, in their current state, will begin being used in Dr. Lori-Ann Forzano's psychology lab at The College at Brockport: State University of New York in the Fall 2018 semester. The Skinner Box 2 program will be used to study self-control and impulsivity in participants varying in age, from pre-school aged children to college students. Based jointly off a physical Skinner Box apparatus used with human participants and a previous version of a Skinner Box program, the current program allows participants to choose between a self-control choice and an impulsive choice to earn video reinforcers. This Skinner Box program is anticipated to be used for years into the future, and resources for future developers have been compiled in the case of updates being made to the program in the future. This program will be used in a multitude of studies performed in the lab, and will be a vital tool for psychological data collection in studies pertaining to the effect of age, reinforcer preference, and reinforcer qualities on self-control and impulsivity. Studies conducted with this program will add to the current research findings regarding self-control and impulsivity.

The application developed to allow participants to select videos from YouTube will also begin to be used with college students in the fall of 2018. Potential participants will be sent this application which they will download and use to select videos from YouTube to use during their session. By allowing each participant to pick their own videos, experimenters can ensure that the participants enjoy the videos and therefore find them reinforcing. Allowing participants to pick their own videos also expands the possibilities for studies that can be developed and conducted, and therefore adds to the current knowledge of the subject of self-control and impulsivity.

References

Charlton, S. R., & Fantino, E. (2008). Commodity specific rates of temporal discounting: Does metabolic function underlie rates of discounting? *Behavioural Processes, 77*, 334–342. doi:10.1016/j.beproc.2007.08.002.

Copyright Law of the United States, Title 17 U.S. Code § 107 (1998). Retrieved from: https://www.copyright.gov/title17/92chap1.html#107

Forzano, L. B., & Logue, A. W. (1992). Predictors of adult humans' self-control and impulsiveness for food reinforcers. *Appetite*, *19*(1), 33-47. doi:10.1016/0195-6663(92)90234-W

Forzano, L. B., & Logue, A. W. (1994). Self-control in adult humans: Comparison of qualitatively different reinforcers. *Learning And Motivation*, *25*(1), 65-82. doi:10.1006/lmot.1994.1004

Forzano, L. B., & Logue, A. W. (1995). Self-control and impulsiveness in children and adults: Effects of food preferences. *Journal Of The Experimental Analysis Of Behavior*, *64*(1), 33-46. doi:10.1901/jeab.1995.64-33

Forzano, L. B., Michels, J. L., Sorama, M., Etopio, A. L., & English, E. J. (2014). Self-control and impulsiveness in adult humans: Comparison of qualitatively different consumable reinforcers using a new methodology. *The Psychological Record*, *64*(4), 719-730. doi:10.1007/s40732-014-0038-7

Forzano, L. B., Porter, J. M., & Mitchell, T. J. (1997). Self-control and impulsiveness in adult human females: Effects of food preferences. *Learning And Motivation*, *28*(4), 622-641. doi:10.1006/lmot.1997.0982

Forzano, L.B., & Schunk, J. (2008). Skinner Box Video [Computer program].

Goodwin, B. C., Browne, M., Hing, N., & Russell, A. T. (2017). Applying a revised two-factor model of impulsivity to predict health behaviour and well-being. *Personality And Individual Differences*, *111*250-255. doi:10.1016/j.paid.2017.02.029

Logue, A. W., King, G. R., Chavarro, A., & Volpe, J. S. (1990). Matching and maximizing in a self-control paradigm using human subjects. *Learning And Motivation*, *21*(3), 340-368. doi:10.1016/0023-9690(90)90013-E

McSweeney, F. K. (2004). Dynamic changes in reinforcer effectiveness: Satiation and habituation have different implications for theory and practice. *The Behavior Analyst, 27*, 171–188.

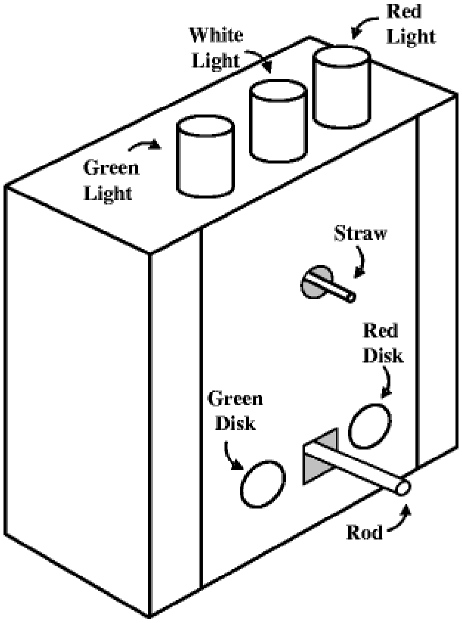
Millar, A., & Navarick, D. J. (1984). Self-control and choice in humans: effects of video game playing as a positive reinforcer. *Learning & Motivation*, *15*203-218. doi:10.1016/0023-9690(84)90030-4

Navarick, D. J. (1998). Impulsive choice in adults: How consistent are individual differences? *The Psychological Record, 46*, 665–674.

Odum, A. L., & Rainaud, C. P. (2003). Discounting of delayed hypothetical money, alcohol, and food. *Behavioural Processes*, *64*(3), 305-313. doi:10.1016/S0376-6357(03)00145-1

Figure 1. Comparison of physical Skinner Box apparatus and original Skinner Box program

a.



b.

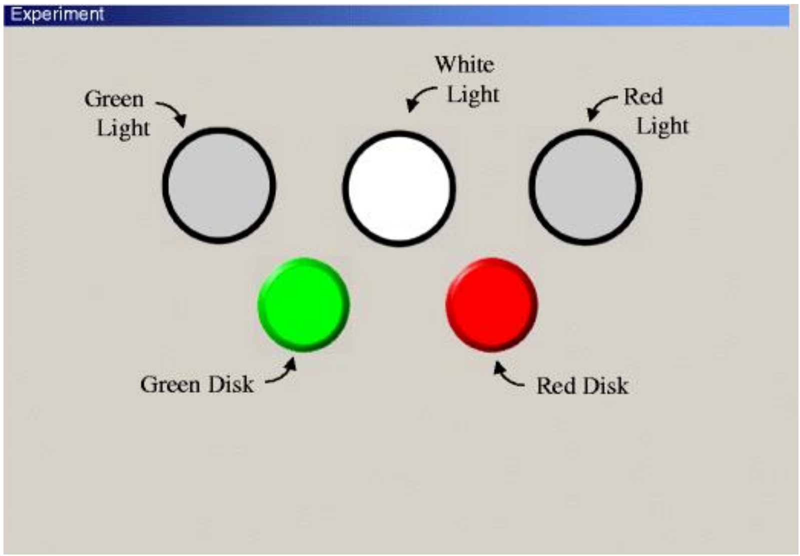
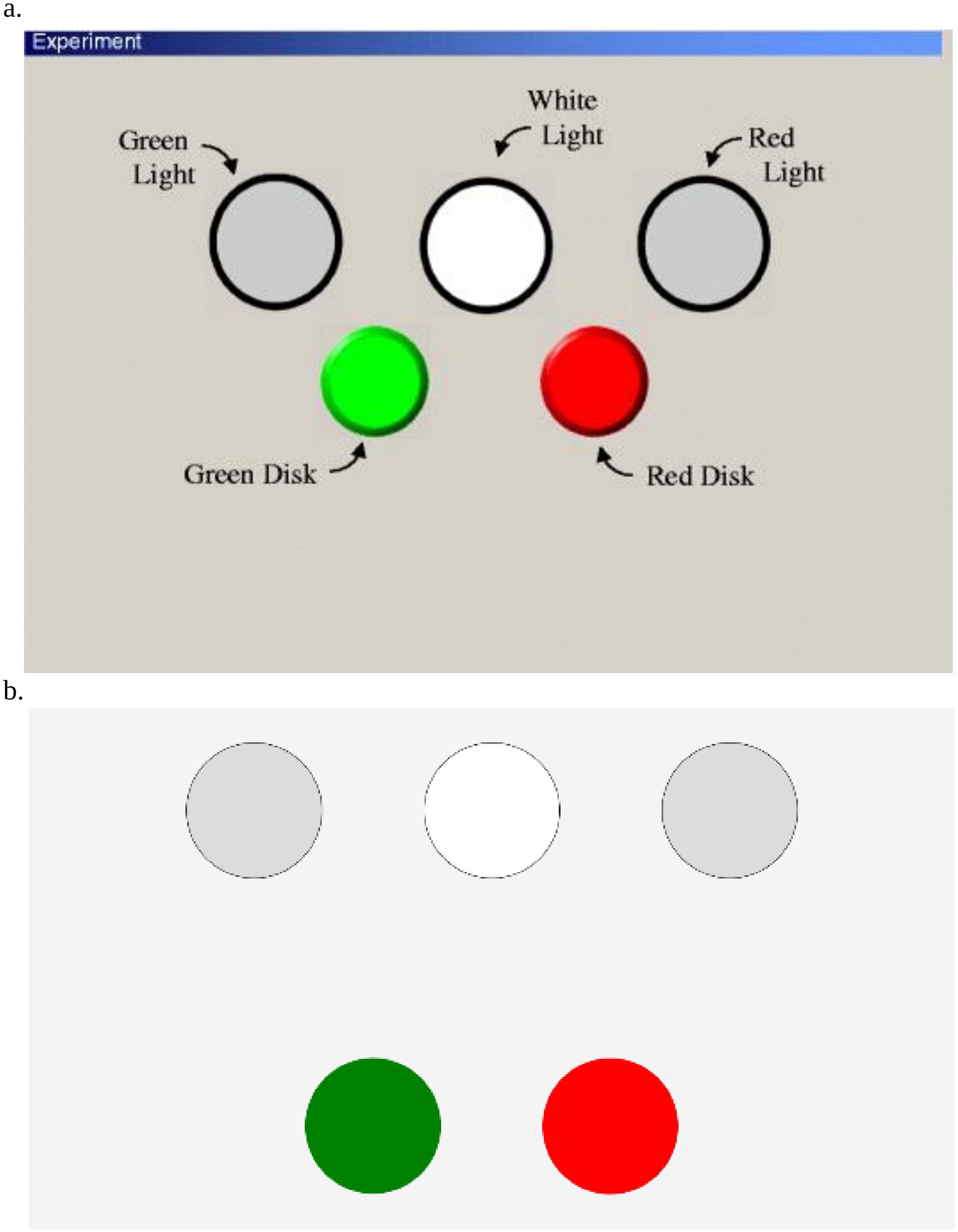


Figure 1. a) Diagram of the physical Skinner Box apparatus used to deliver juice as a reinforcer (Forzano et al., 2014). b) Diagram of user interface of original Skinner Box Program (Forzano et al., 2014). The program interface closely replicates the physical box, with two delay lights and a trial light at the top and two selection disks. They differ in participant selection, as the physical apparatus has a rod that is pushed to a side for selection, and the computer program uses clicking on the disks themselves as selection.

Figure 2. Comparison of user interfaces in original and new Skinner Box programs.



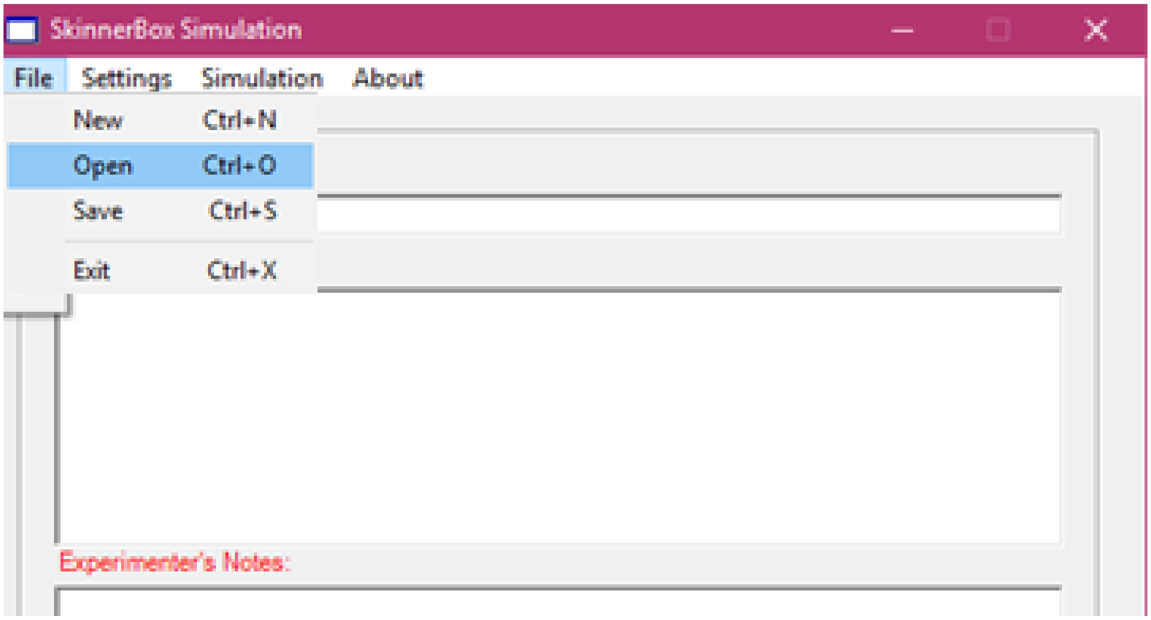
a.

b.

Figure 2. a) Diagram of user interface of original Skinner Box Program (Forzano et al., 2014). b) Screenshot of the new Skinner Box program during execution. The user interface of the new program was designed to replicate the original user interface as closely as possible.

Figure 3. Comparison of input selection in previous and current Skinner Box program

a.



b.

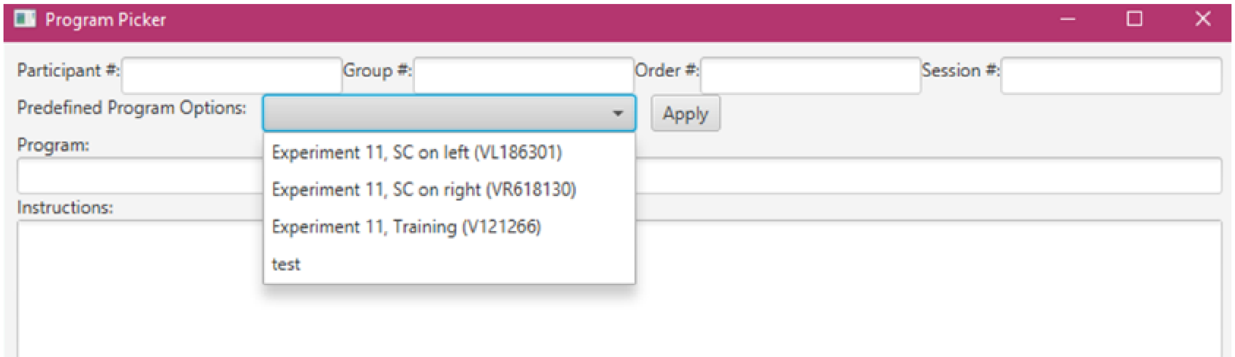
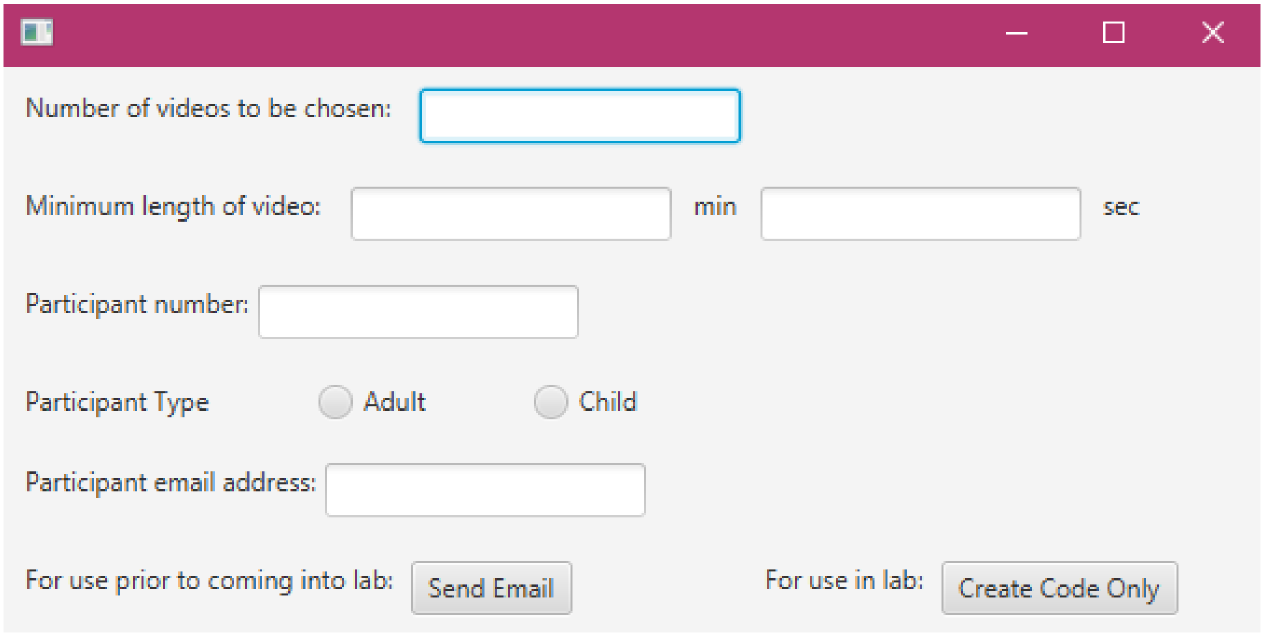

Program Picker
Participant #:
Group #:
Order #:
Session #:
Predefined Program Options:
Apply
Program:
Instructions:
Experiment 11, SC on left (VL186301)
Experiment 11, SC on right (VR618130)
Experiment 11, Training (V121266)
test

Figure 3. a) Screenshot of input selection for original program. Selecting the 'Open' option opened a file browser in an additional window to browse the computer for the appropriate input. b) Screenshot of input selection in new program. The options are available in the drop down screen, leading to an easier selection process.

Figure 4. User interface for entering data for the generated code



Number of videos to be chosen:

Minimum length of video: min sec

Participant number:

Participant Type ○ Adult ○ Child

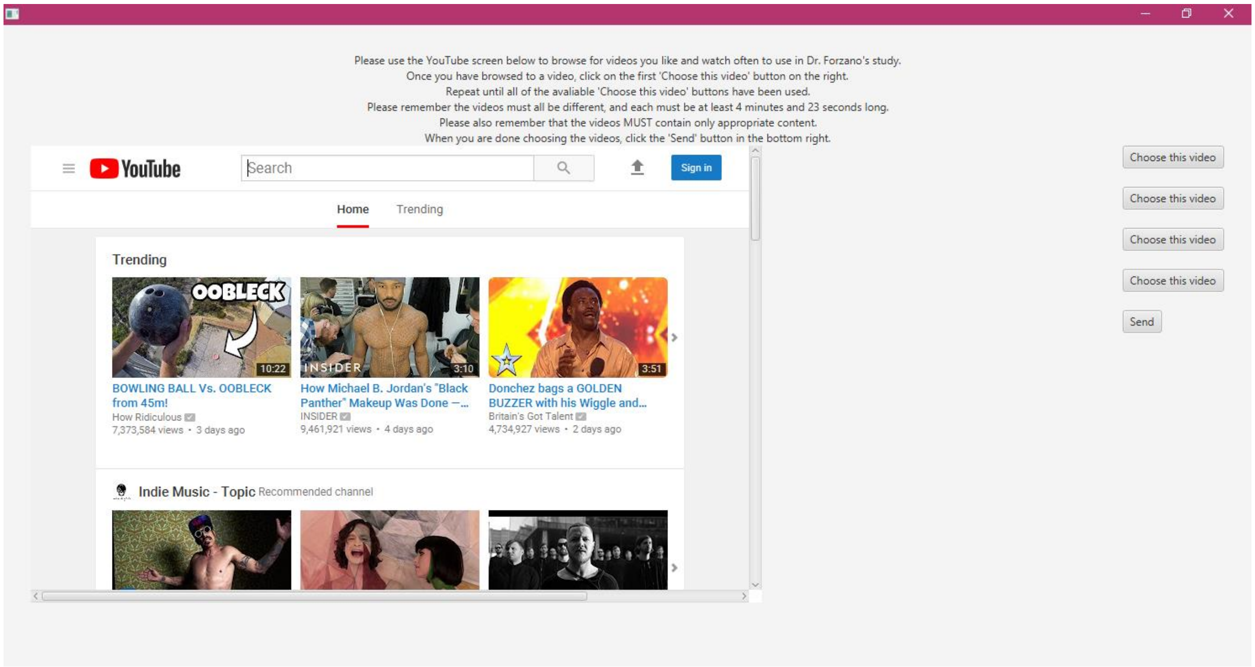
Participant email address:

For use prior to coming into lab: Send Email

For use in lab: Create Code Only

Figure 4. Screenshot of screen for entering study and participant information. When ‘Create Code Only’ is selected, the code is generated and displayed. When ‘Send Email’ is selected, the code is generated and placed in the email which the application then sends to the entered participant email address.

Figure 5. User interface for selecting videos



Please use the YouTube screen below to browse for videos you like and watch often to use in Dr. Forzano's study.
Once you have browsed to a video, click on the first 'Choose this video' button on the right.
Repeat until all of the avaliable 'Choose this video' buttons have been used.
Please remember the videos must all be different, and each must be at least 4 minutes and 23 seconds long.
Please also remember that the videos MUST contain only appropriate content.
When you are done choosing the videos, click the 'Send' button in the bottom right.

YouTube
Search
Sign in
Home
Trending
Trending
OOBLECK
10:22
BOWLING BALL Vs. OOBLECK from 45m!
How Ridiculous
7,373,584 views • 3 days ago
INSIDER
3:10
How Michael B. Jordan's "Black Panther" Makeup Was Done —...
INSIDER
9,461,921 views • 4 days ago
3:51
Donchez bags a GOLDEN BUZZER with his Wiggle and...
Britain's Got Talent
4,734,927 views • 2 days ago
Indie Music - Topic Recommended channel
Choose this video
Choose this video
Choose this video
Choose this video
Send

Figure 5. Screenshot of screen for selection of videos. Instructions are displayed at the top for how to use the application and what videos can and cannot be selected. The embedded YouTube page can be used to browse to the desired videos. The ‘Choose this video’ buttons are used to select a video, provided it meets the specified criteria. The participant then sends their selection to the lab using the ‘Send’ button.

Appendix A

Skinner Box 2 Informational Documentation

May 11, 2018 – Revision 3

Development Information

1. The Skinner Box 2 program was written by Cheryl Limer in the 2017-2018 school year for an honors thesis. It is written in Java, and the source code can be found on a flashdrive Dr. Forzano has with all of the information regarding the thesis. The source code can also be found in a zip drive on the Forzanolab@gmail.com google drive, as well as in an email sent to Dr. Forzano in May of 2018. As of May 2018, Cheryl can be reached at limerca@gmail.com
2. The Skinner Box 2 program was based off a program, sb.exe, written in 2008 in C# by Jason. As of May of 2018, this program is not compatible with Windows 10 and can only be run in a Virtual Machine running Windows 8 or below due to the depreciated DirectX package used to control video playback.

Putting the Skinner Box 2 program on a new computer

1. Locate the 2018_New_Skinner_Box folder which can be found on the flashdrive, or on the computer in the chamber, the computer in the back room, or Laptop 2 in the C drive. It can also be found on the computers in the back room and cubicle 3 in the folder location C:/Kids Study/Experiment #11 Fall 2017 – Children.
2. Once this folder is located, make sure that the following files and folders are contained within the SkinnerBox_2 folder:
    a. Skinner Box 2 final.jar
    b. Presets (folder)
    c. Videos (folder)
    d. Results (folder)
3. If all of the listed files and folders are in the SkinnerBox_2 folder, copy this whole folder to the new computer that you would like to run participants on via flashdrive. This folder must be put in a location in which there are no spaces in the names of any folders used to get to the application. For this reason, and so it is easy to find, put the SkinnerBox_2 folder in the C drive.
4. The program should then be functional on the new computer. Check this by attempting to run the program.
5. If it does not open, or if something is not working correctly, double check that you followed the steps above. Most importantly, make sure that the folders listed above are in the same location as the program itself, and that all of the folders used to get to the program do not contain any spaces in their name.
6. Before running participants, ensure that the appropriate videos for the study are in the Videos folder and do not have spaces in their names, and that presets have been created

to autofill the setup screen. Before beginning a new study, it is also recommended that you empty the Results folder associated with the program, but first make sure all of the results in that folder are saved in a different location. Do not create subfolders for different studies within the Results folder associated with the program, as the program will only write results in the Results folder and not created subfolders.

Creating and Maintaining Presets (Please note: adding or deleting presets only adds or deletes them on the specific computer you perform these steps on, NOT all computers)

1. The files contained in the 'Presets' folder are used to autofill all of the information on the setup screen of Skinner Box 2, including name, instructions, delay and amount period times, etc. There should be a preset created for each session of the study being run.
2. To save a new preset, open the Skinner Box 2 program. Enter in the information you wish to be saved. All of the areas must be filled, with the exception of the areas pertaining to participant information.
3. The information entered under 'Program' is how these entered values will appear in the dropdown in the future. The information entered under 'Instructions' is what will appear to the participant before they click the start button. (If, when testing this preset, the instructions get cut off, try to make the lines cut off shorter by putting some of it on a new line). The information entered under 'Experimenter's Notes' is for the use of the experimenter and will only be seen on this screen.
4. When choosing a reward type, 'video' means the video being used in the session is one that we always have converted and in the lab. 'youtube' means the video was chosen by the participant from youtube, converted, and put on the computer. If 'youtube' is selected, the video will be deleted upon completion of the session.
5. Please double check all of the information entered on this screen as it will be what is used every time a participant is run.
6. To save this information, click the 'Save entered values as preset' button at the bottom. The next time the program is opened, this new preset will be able to be selected from the dropdown.
7. Check the preset by continuing setup, choosing a video, and viewing the instruction page to ensure the instructions are completely visible on the screen.

Deleting presets from previous studies (Please note: adding or deleting presets only adds or deletes them on the specific computer you perform these steps on, NOT all computers)

1. In the case that a new study is being run with new preset values, it may be helpful to delete the presets of previous studies to minimize the options in the dropdown list.
2. Before you delete anything, go into the 'presets' folder and copy the files that are from previous studies. Save these files on the computer in the back room in the folder corresponding to the study that they are from.
3. Once you have saved them, you can delete the files from the presets folder. They will no longer appear in the dropdown list when you used the program.

Unique Skinner Box 2 Features

1. Back to Back sessions: When this checkbox is selected on the setup screen two sessions are run back to back. That means two videos are chosen, one for the first session and one for the second session, and the delays and reinforcer lengths switch sides. For example, if the self-control choice was on the left side for the first session, it would be on the right for the second. This function should not be used with training sessions where the delays and reinforcer lengths are the same on both sides. The results of both of the back to back sessions will appear, separated, on the same results document.
2. Youtube Function: In terms of setup and from a participant's perspective, this runs the exact same way that the 'video' option does. The only difference is that at the end of the session, when the screen asking the participant to get the experimenter appears, the application deletes the video that was in use. This should be used with videos that had been downloaded and converted from YouTube, and should not be used with videos that were converted from DVDs and are on the computers to use multiple times.

Appendix B

Fair Use Policy of Copyright Law of the United States (1998) cited in its entirety

“Notwithstanding the provisions of sections 106 and 106A, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright. In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include—
(1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
(2) the nature of the copyrighted work;
(3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
(4) the effect of the use upon the potential market for or value of the copyrighted work.
The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors.”

Appendix C

Video Selection and Preparation Protocol

YouTube and DVD cartoons

May 11, 2018 – Revision 5

Section 1 - To be used when an email is being sent to the participant asking them to choose youtube videos before coming in for the study

Opening the YouTube Application

1. The application used to send the email to participants can be found in two places. It is on the computer in the back room and the computer in cubicle 3, in the file location C:/Kids Study/Experiment #11 Fall 2017 - children/2018_New_SkinnerBox and is called Youtube Application final.exe
2. The application can also be downloaded from the internet and used on any Windows computer with Java installed. All of the on campus computers fit this criteria. To download from the internet, log on to the lab email account, with the email forzanolab@gmail.com and the password 'Fall2017!!'. After you are logged in, click on the icon with the little squares in the upper right hand corner, and then click on google drive. Once there, click on and download the application Youtube application v5.exe. You can then run it by going to your downloads folder and clicking on the application. If you are having issues downloading, try signing out of all of your other gmail accounts and logging back into the lab email.

Sending email to Participant to pick YouTube videos

1. Go to 'YouTube Application v3' and double click on it to open it. If you downloaded the document, it may ask you to save the file. To open it after a download, go to your downloads folder and double click on it. A blue box may appear asking you to enter your username and password. Entering your netid username and password here will allow the program to run.
2. Where it asks for a one time use code, enter 'admin' and click the Enter button.
3. In the first text field, enter in the number of videos that will be needed for the entirety of the study, with one video for each regular session and two videos for each back to back session. So, in Experiment 11 for example, there are 3 regular session, and therefore 3 videos are required.
4. In the next text field, enter the minimum length of video needed. This number is found on a calculator using the following equation: ((*number of free choice trials* +4)* *longest reinforcer length in seconds*). Divide by 60 to find the number of minutes, and use (*the remainder* * 60) for the number of seconds. So, in Experiment 11 for example, there are 10 free choice trials and the longest possible reinforcer amount is 18 seconds, so you would do (10+4) * 18 seconds, which equals 252 seconds. Dividing by 60 equals 4.2.

Therefore, the minimum number of minutes would be 4, and the minimum number of seconds would be (.2 * 60) which equals 12 seconds. If either the minutes or seconds is 0, please enter 0 and do not leave any field blank.

5. In the next text field, enter the participant's participant number.
6. Using the selector, indicate whether the participant is an adult, ie choosing the videos for themselves, or whether the participant is a child, ie the email is being sent to a parent who will be choosing the videos for the child.
7. Finally, enter the participant's email address.
8. When you click the 'Send Email' button, the information entered will be used to be make a code which will be send via email to the participant with a link to the application. Click the 'Exit' button to close the application.
9. If you only want to generate a code but do not want to send an email, in the case that the application is being used by the participant in the lab, rather than before coming into the lab for the session, click the 'Create Code Only' button. This would be best to do right before the participant arrives in the lab a laptop with internet access that the participant will use to select the videos. After the code is generated, copy it, and open up the application again (see step 1). Paste it into the 'one time use code' section, and click 'Enter'. The application will then be ready for the participant to select the videos. Once the videos are selected by the participant, begin the accessing participant emails protocol below.
10. If reminder emails need to be sent, this can be done by logging into the lab email and writing and sending the participant an email them reminding them to use the application and pick the videos at their earliest convenience, or a new email with the link can be sent by following this protocol a second time, starting at the beginning of this section of the protocol.

Where Information is Kept Within the Code

1. The information entered above is all contained within the generated code at specific places. The format of the code is as follows, where the description in parenthesis describes the information the parenthesis would be replaced with: 'P'(participant number)('A' is the participant will be choosing the videos for themselves or 'C' if the participant is a child with a parent choosing videos for them)(5 random characters)'T'(minimum minute length necessary for a video)'M'(minimum second length necessary for a video)'S'(number of videos to be chosen).
2. Given the example code P32AD34HST2M7S3
   a. 32 is the participant number, as it comes after the letter P at the beginning and is all of the numbers before the letter A
   b. A indicates that the participant is considered an adult participant and not a child participant
   c. D34HS are all purely random characters and hold no meaning
   d. T is the last T in the code, and therefore indicates that what follows is the minimum length of a video

   e. 2M7S following the T means that the video must be at least 2 minutes and 7 seconds long
   f. 3 indicates that the participant will need to select a total of 3 videos for the entire study
3. This information is used by experimenters when determining the participant number from the code. The rest of the information is used by the YouTube application to set up the user interface with the correct number of select video buttons, correct instructions, and appropriate video filtering.

To Access Participant Emails

1. Once the participant has completed the selection of their videos, an email will be sent to forzanolab@gmail.com. The password for this account is 'Fall2017!!' This password can be changed by using Google's functions for changing passwords, either through the 'I forgot my password' function, or through settings.
2. You will have to check the inbox of this email account until the email with the participant's code arrives. The participant number is found within the code after the letter P but before the next letter. So if the beginning of the code was P45A6…, the participant number would be 45. Once the email arrives, open the email and convert and download the videos.
3. To do this, go to https://www.onlinevideoconverter.com/video-converter. It will probably be helpful to have two web browsers open and split across the computer screen.
4. Enter the first youtube link in the area labeled 'Paste link here'
5. In the area directly below, use the dropdown to change the format to .mp4
6. Click the orange 'Start' button and your video will begin to convert. Note: This website may have pop-up ads. If anything comes up on the screen during this process, close out of it without interacting with it and the conversion process will still be happening in the background. Once it is converted, click the button next to the file that says 'Download' to download the file to the computer. This will save it in the 'Downloads' folder on the computer. If it asks you to open or save the file, choose the option that says open file location, which will take you to the downloads folder. Otherwise, use file explorer to go to the downloads folder. Once you are in the downloads folder, rename the file (by right clicking on the file and selecting 'rename') with the participant number (again, found after the letter P in the associated code from the email) and the number video it is with an underscore underneath. So, for example, if it is the first video you have converted for participant 99, label the video 99_1.mp4
7. Repeat this process for all videos, until all the videos are converted, and appropriately labeled.
8. Once the videos are all downloaded and renamed, use the flashdrive for the study to move the videos to the computer the participant will use during the study, in the case of Experiment 11, both the chamber computer and laptop 2. Put the video in the Videos folder found in the SkinnerBox_2 folder found on the C: drive. Once the videos have been put in the Videos folder on the computer, delete the videos off the flashdrive. Deleting the files keeps the space taken up on the flashdrive to a minimum, and also

helps adhere to copyright law by making sure each downloaded video is only watched once during the session and then deleted.

Section 2 - To be used when converting a DVD to mp4 files on the computer

*note: This must be done on a computer where you hold administrator privileges and can download applications

Finding appropriate software

1. DVD's are encrypted so that it is difficult to save the content of a DVD to a personal computer. To get around this, you have to use software that can crack this encryption, and because this technically goes against copyright laws (even though we are using the videos in a way that does not violate copyright law), software like this can be hard to find.
2. There is a converter that can be installed saved on the computer in the back room, however it is not clear if it will work with more recent DVD encryption. It is possible to find free trial downloads of some software on the internet that will work for long enough to convert at least some DVDs. To find these you will need to google search for free DVD to mp4 converters. Some free trials are very small, but some are long enough.
3. Once you find software that you think will work, download it to your computer.

Converting the DVD

1. Depending on the software you are using, this will vary. In general, you will put the DVD in the computer, upload it to the software, and have the software convert it to an mp4. It will likely save it to a downloads folder or a folder associated with the software.
2. When converting, you will likely be able to convert only one section of the DVD at a time. Convert one episode of a time, excluding the theme song so that the beginning of the video is the beginning of the actual episode.
3. Once it is converted, it should be renamed based on the same of the cartoon and which one of the series it is with an underscore underneath. So, for example, if it is the first SpongeBob episode, it should be labelled spongebob_1.mp4. The videos should then be moved via flashdrive to the computer on which the study is being conducted to be saved in the Videos folder.

Appendix D

YouTube Application Informational Documentation

May 11, 2018 – Version 1

Development Information

1. This application was created by Cheryl Limer in the 2017-2018 school year as part of an honors thesis. The application is written in Java and has been exported as a .jar file and converted to an .exe file so that it can be downloaded by potential participants.
2. The source code can be found on a flashdrive Dr. Forzano has with all of the information regarding the thesis. The source code can also be found in a zip drive on the Forzanolab@gmail.com google drive, as well as in an email sent to Dr. Forzano in May of 2018. As of May 2018, Cheryl can be reached at limerca@gmail.com

Notable Features

4. ‘One Time Use Code’: This application generates a code for a participant in order to associate a participant with their video selection without infringing upon confidentiality rights. There is also information necessary for the application to run embedded in the code in the following way where the description in parenthesis describes the information the parenthesis would be replaced with: ‘P’(participant number)(‘A’ is the participant will be choosing the videos for themselves or ‘C’ if the participant is a child with a parent choosing videos for them)(5 random characters)’T’(minimum minute length necessary for a video)’M’(minimum second length necessary for a video)’S’(number of videos to be chosen).
   a. Given the example code P32AD34HST2M7S3
      i. 32 is the participant number, as it comes after the letter P at the beginning and is all of the numbers before the letter A
      ii. A indicates that the participant is considered an adult participant and not a child participant
      iii. D34HS are all purely random characters and hold no meaning
      iv. T is the last T in the code, and therefore indicates that what follows is the minimum length of a video
      v. 2M7S following the T means that the video must be at least 2 minutes and 7 seconds long
      vi. 3 indicates that the participant will need to select a total of 3 videos for the entire study

   The program extracts this information from those specific places in the code in order to ensure the correct number of videos is selected, the videos are the correct length, and that the instructions are appropriately varied for an adult or child participant. This code is also generated by an administrator user interface which can be accessed by typing “admin” into the “one time use code” text area. The necessary information is entered by the experimenter, and if an email is to be send to the participant, clicking the ‘send email’ button will generate the code, put it in an email, and send it to the participant. If just the code is to be generated, in the case that the participant will be using the application in the lab at the time of participant rather than on their own before participation, the 'Create Code Only' button can be used.

5. Video Selection Restrictions: The application restricts the selection of certain videos due to uniqueness, content, and length. Before a video can be selected, the application makes sure that it is not the same as any video already selected. It is also restricted based on content. YouTube makes certain inappropriate videos unavailable to view unless logged in to prove the view is at least 18 years old. These videos are tagged with a certain content tag. The application ensures that the videos do not have this content tag before it can be selected. Finally, the application uses the minimum length found in the code to make sure any video selected is at least that long. If the video is not that long, it cannot be selected. If a video fails to meet any of these criteria, an error box pops up letting the participant know why that video cannot be selected and a new video must be selected. Information on how this is done can be found in the 'Technical Specifications' section below.
6. Email: This application sends emails in two specific situations. When an experimenter is creating a code, the application can send an email to the entered participant's email with the generated code, instructions, and a link to the application for download. The text of this email can be changed by altering the source code. Information concerning this is in the 'Technical Specifications' section of this document. The application also sends an email to the forzanolab@gmail.com email account after the videos have been selected with the code containing the participant number and the youtube links for the videos to be converted and downloaded. A flowchart outlining this process can be found as the last page of this document, and more details on how this is done can be found in the 'Technical Specifications' section of this document.

Streaming and Copyright Considerations

1. It was decided to not stream videos directly from YouTube at time of participant for the following reasons
    a. Streaming videos does not allow for the required amount of control for timing of the program
        i. There is a long delay the first time the video is loaded
        ii. There is a shorter but not necessarily consistent delay each time the video begins again
        iii. There is no way to start from the exact place the video was stopped, so if the delay is slightly longer or shorter than expected the video may skip a section
        iv. Speed of load/play back depends on others' Wi-Fi usage
    b. Some videos cannot be embedded so there would need to be requirements as to what videos could be chosen, or a non-embeddable video would be viewed with the entire webpage including video controls, comments, and similar videos.

   c. Some videos have ads, and while it seems like the ad might only play the first time the video is loaded and could be taken care of before the actual trials, an ad that played would undermine the entire trial and they cannot be completely avoided.
2. Downloading the chosen videos before the participant arrives completely eliminates ads, gives the required control over the loading, timing, and playback of the video, and ensures all videos can be viewed embedded with no other options for the participant but to watch the video
3. It was verified that this is within the bounds of copyright laws for the following reasons
   a. Streaming videos from YouTube is always legal. Even if the video is violating copyright laws by being on YouTube, it is illegal for the person who uploaded but it is completely legal to watch it on YouTube.
   b. Some videos on YouTube are not copyrighted and it is completely legal to download, convert, and use the converted video as much as you would like. These videos are ‘copyright free’ or ‘royalty free’ videos.
   c. In our case, the copyrighted videos fall under the Fair Use policy in copyright law, Title 17 U.S. Code § 107, cited at the end of this document with the most relevant parts highlighted. We are using these videos in short segments for educational and research purposes, and therefore our use of them is covered under the Fair Use provision.
   d. We will also be downloading the video from YouTube, giving the video an accurate amount of views on the website and seeing the ads that are there. It will be watched once only in downloaded form and then be deleted. This mimics watching the video on YouTube, and therefore should not be violating copyright law to begin with, Fair Use provision notwithstanding. There have been court cases involving YouTube and conversion websites, where is has been determined that the conversion website was allowed to exist provided the videos were not archived after conversion.
4. Our decision not to stream the videos at time of participation is the reason that this YouTube application is necessary, so that the videos can still be selected from YouTube but are converted and downloaded to eliminate the issues related to streaming.

Download Issues and Possible Solutions

1. In its current form, this application is an .exe application uploaded to the lab’s google drive. Participants are given the sharable link to this application, and they are to download and use the application on their own time before coming into the lab.
2. This application requires Java to be downloaded on the computer in order to run. This means that the application will not work on Apple computers, nor will it work on Windows computers without a version of Java. Due to different configurations on personal computers, there is a significant proportion of potential participants who cannot download and use this application on their personal computers.
3. For participants that are college students, we have requested in the instructional email that they use a campus computer to use the application because all campus computers are

Windows computers with Java downloaded. They are given specific instructions on how to download and use the application based on how campus computers are configured.

4. For participants that are not college students, namely child participants, we cannot assume they have access to a campus computer. Therefore, child participants will use this application in the lab during the first session. Before the participant arrives, the experimenter will generate the code and enter it into the application so that the video selection screen is ready for the participant. After the consent form is signed, the participant will selected the necessary videos and the experimenter will need to download and convert the first one before the trial can begin. This is not a particularly optimal solution, but it is workable.
5. A solution in progress as of May 2018 is to have a computer science student redevelop this application as a purely web application that can be run exclusively in a browser with any configuration as an independent study project.

Technical Specifications

1. Changing the text of the email: The text of the email is created in the createBody() method of AdminGUI class. It is all hardcoded with the exception of the minimum length of the video, which is added through variables holding that information. There are two versions of the email, one for if the participant is an adult, or college student, in which the directions are directed at the participant. The other version of the email is used for when the participants are children, in which case the email is directed at the parent of the participant.
2. Changing the text of the instructions on the video selection page: The text of the instructions that are displayed on the video selection screen the participant sees is hardcoded and put in a Label object in the start() method of the GUI class. The minimum length of video and language directed at either the participant or parent of participant are added to the hardcode through variables holding that information.
3. Re-exporting application after change to source code: The application can be exported as a .jar file by File->Export and then selecting the ‘Runnable Jar File’ option. It should be named Youtube application vx where the x is the next version number. Once the .jar is exported, it can be converted to an .exe using wrapper software such as Launch4j. The .exe would then be uploaded to the google drive of the forzanolab@gmail.com account in the same place as the previous version. This can be done by going to the current one that is there, right clicking on it, choosing ‘Manage Versions’ and then clicking ‘Upload new version’. **NOTE: This application needs to be put in the same place as the old one was, or the sharable link in the email will still take the participant to the old version and not the new one.** If a new version is just uploaded to the drive but is not added as a new version of a previous application, the sharable linkn will be different. Because this link is hardcoded, the new version of the application has to go in the same place as the old one. The source code could be changed to go to a new link, but if you update it to include a new link, the newest version with the new link hardcoded must go to whatever link is in the code.

4. Interactions with YouTube API: The YouTube videos themselves are determined by a JavaFX webengine object which can detect a change in the URL of the embedded page. Because YouTube uses AJAX however, the page must be refreshed in order for the URL change to be detected by the webengine. When the user presses the button to select the video, the webpage is refreshed and the URL can be retrieved. The URL is what is sent to Dr. Forzano's lab once all of the videos are selected. The URL of a youtube page also contains the video ID, which is parsed out and saved.

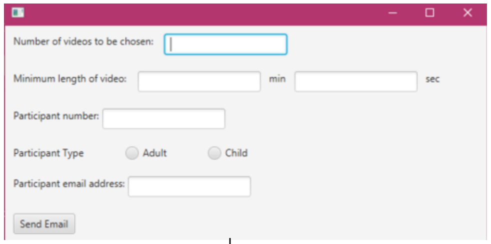
   Google releases APIs and jar files that can be used by the general public to interface with google products. In order to use these API's, you must register with google through a valid gmail account. forzanolab@gmail.com was used to register and get the key that is used for this application. The key is associated with this specific project and the YouTube API. This key can be managed through the Google API Development Console (https://console.developers.google.com/apis/dashboard?angularJsUrl=%2Fapis%2Fdashboard&authuser=1&_ga=2.66384365.-2132813164.1521044727&project=skinner-box-199615&folder&organizationId&duration=PT1H). Using this key and sample code, the video ID is passed in methods to determine the length of the video and the content tag. The content tag is null if the video is available to be watched by anyone, and has a string if YouTube thinks you have to be 18 to watch it. The code for this, adapted from the sample code found online, can be found in the videoParse() method of the GUI class.
5. Interactions with Gmail: The application uses JavaMail to send the emails to the participant and also from the video selection screen to the lab. JavaMail requires a hardcoded username and password of a gmail account that it can log into and send the email from. This gmail account must have the setting enabled that allows external applications to access it. Originally, it was thought that the emails could be sent from the forzanolab gmail account, but because the code needs the username and password, it means that any time the forzanolab password changed, the source code would need to as well. The application also couldn't read the username and password from an editable file, as the application is getting downloaded and used on many computers and that file would need to also be downloaded every time the application was, which is tedious and also gives everyone the forzanolab login information. As a solution, a dummy email account was made. This email account is only used to send emails from the application, and the source code for this application is the only place the login information for this account is. This login information therefore should never need to be changed, and the application should always be able to send emails from it. The application uses JavaMail to send the emails in the Email class.

Copyright law, Title 17 U.S. Code § 107

Notwithstanding the provisions of sections 106 and 106A, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright.

In determining whether the use made of a work in any particular case is a fair use the factors to be considered shall include—
(1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
(2) the nature of the copyrighted work;
(3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
(4) the effect of the use upon the potential market for or value of the copyrighted work.
The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors.

No.reply.psychlab@gmail.com



Number of videos to be chosen:

Minimum length of video: min sec

Participant number:

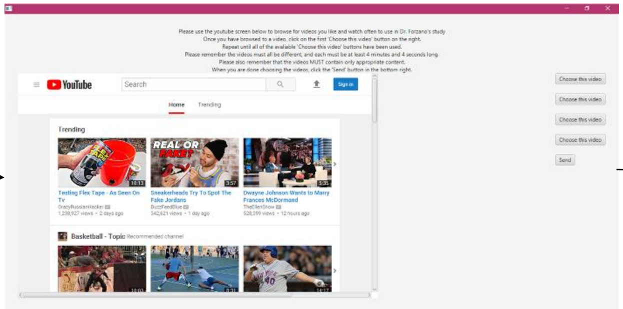
Participant Type ○ Adult ○ Child

Participant email address:

Send Email

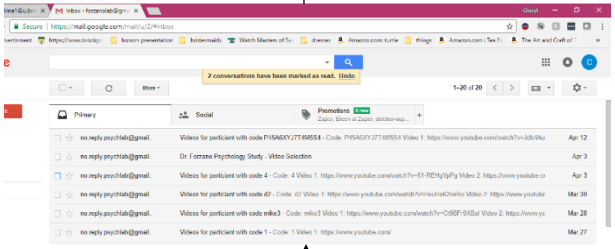
Application logs into and uses ‘no.reply’ email account to send the email to the participant with their code and the link to the application

Participant follows link to Google Drive, downloads application, uses it to pic YouTube links



Please use the youtube screen below to browse for videos you like and watch often to use in Dr. Forzano's study
Once you have browsed to a video, click on the first 'Choose this video' button on the right.
Repeat until all of the avaliable 'Choose this video' buttons have been used.
Please remember the videos must all be different, and each must be at least 4 minutes and 4 seconds long.
Please also remember that the videos MUST contain only appropriate content.
When you are done choosing the videos, click the 'Send' button in the bottom right.

Application logs into and uses ‘no.reply’ email account to send an email to the ‘forzanolab’ email account with the code and chosen links



Inbox - forzanolab@gm
https://mail.google.com/mail/u/2/#inbox
2 conversations have been marked as read. Undo
More
1–20 of 20
Primary
Social
Promotions

no.reply.psychlab@gmail.. Videos for particiant with code P45A6XYJ7T4M5S4 - Code: P45A6XYJ7T4M5S4 Video 1: https://www.youtube.com/watch?v=Jdb4Au Apr 12

no.reply.psychlab@gmail.. Dr. Forzano Psychology Study - Video Selection Apr 3

no.reply.psychlab@gmail.. Videos for particiant with code 4 - Code: 4 Video 1: https://www.youtube.com/watch?v=51-REHgYpPg Video 2: https://www.youtube.c Apr 3

no.reply.psychlab@gmail.. Videos for particiant with code d2 - Code: d2 Video 1: https://www.youtube.com/watch?v=HxuHo62reRw Video 2: https://www.youtube Mar 30

no.reply.psychlab@gmail.. Videos for particiant with code mike3 - Code: mike3 Video 1: https://www.youtube.com/watch?v=Ct9BFr9XBal Video 2: https://www.yc Mar 28

no.reply.psychlab@gmail.. Videos for particiant with code 1 - Code: 1 Video 1: https://www.youtube.com/ Mar 27

Videos can then be downloaded and converted for use during the session