After pondering ideas for my final project, I have decided to create a trivia game called “I’m not a robot”. I was inspired by the “reCAPTCHA” system that enables web hosts to distinguish between human and robotic input. In a whimsical way, I imagine the verification process including a series of puzzle games that the system tests against the user to ensure they are real. The puzzles will include a series of objectives that the user must successfully answer within the depicted time limit. Therefore, at this stage, the game is to be described as follows…

As the user starts the simulation a small input field, that looks like a “reCAPTCHA” window will appear in the middle of the screen. The user will instinctively click on I’m not a robot, and an “ERROR” will appear, with patterned virus bugs appearing all over the x and y-axis. After a few seconds the state will automatically change to the title / instructions and provide the user with further explanation as to what they just experienced, and what they must do to prove that they are not a robot. Once the user reviews the instructions, the simulation will officially start, and the user will then be guided to the first of 3 states where they must successfully pass each objective. A single failure will lock the simulation and put the user back to the start… it’s a security measure.

The first puzzle, also known as the “audioPuzzle” will take advantage of the audio in class. Since my program resides around the idea of proving oneself that they are in fact human, I figured the more interactive components the better. The premise of this puzzle is that the user must navigate the vehicle from the bottom of the canvass towards to the top solely with their voice. The louder they speak, the faster the car moves along the y-axis. If they speak quietly the car moves slowly, and if they don’t speak at all, the car starts to reverse back to its starting position on the y-axis. This on its own would be too simple, so I will add road obstacles such as potholes. However, these potholes will translate upon the x-axis, because why not? We all know how potholes like to play games with us.

The second puzzle, also known as the “ballPuzzle”, will take advantage of keyboard input. In a traditional, non-traditional, the user will need to bounce balls upon an apparatus to collect tokens. They will only have so many balls, so they will have to ensure that they can achieve all the tokens. They will jitter along the x-axis, while numerous balls fall from the top of the canvas. I plan to use a PolySynth and create a scale so that once a ball touches the apparatus a random note will play. I may add a delay to create a reverberation effect, so that as the game progress it becomes more difficult since the user player will hear the delay of all the bounced balls. This puzzle fits within the theme, as a computer would have trouble deciphering a random element and intended element simultaneously within a short amount of time.

The third puzzle, in state 3, is less of a puzzle and more of a verification. With the use of the user’s webcam, the system will scan the users face and will only proceed if it can be matched with one of a human. Horizontal and vertical bars will animate on each axis, while the scan is being done. If the scan is unsuccessful, like in a sense that no webcam is detected, or that the image in front of the camera is not human, the simulation will fail. If the scan is successful, the simulation will lead to its’ final stage, that appears once again the “reCAPTCHA” looking window, however this time when the user clicks on the “I’m not a robot”, a final state will appear that says something along the lines of “you are not a robot”.

The largest challenge with this conception is the use of the webcam. After briefly speaking with you, I have decided to use the ml.js library to implement face detection. Once I implement the initial code and have the webcam working correctly, I look forward to learning how I can take advantage of it and what I can make from it to help bring together the simulation.

The program will end once the facial recognition successfully states that the object in front of the camera is at least 85% human…. A final state will appear that simply says “Indeed, you’re not a robot”.

States with various puzzles

1. Maneuver the car around the potholes with your voice
2. Collect all the tokens before the balls disappear
3. Scan face with webcam to prove identity

**Notes**Prototype is just to include shapes. Don’t bother with images yet.

Various timers will be used throughout the program to assist with the story telling aspect. This allows the user to become better engaged with the program. The creation a smooth flow develops from the introduction, to the 3 “tests”, while finishing with the conclusion.