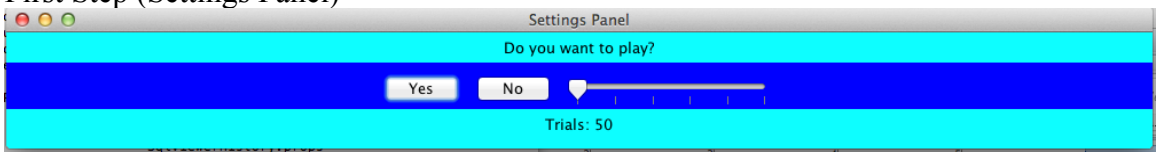
Matthew Vollkommer 810249412 P3

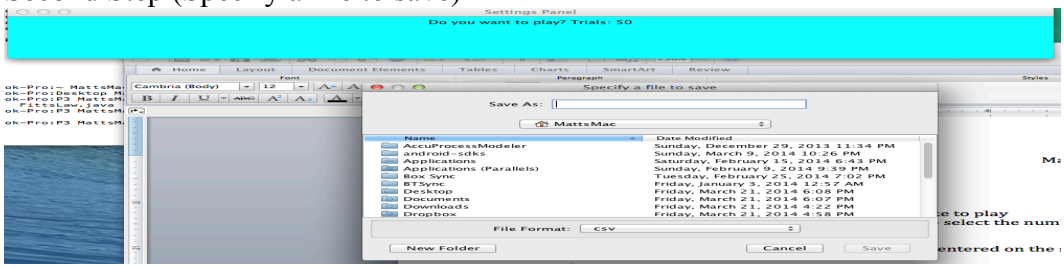
User Manual

1. First Step (Settings Panel)

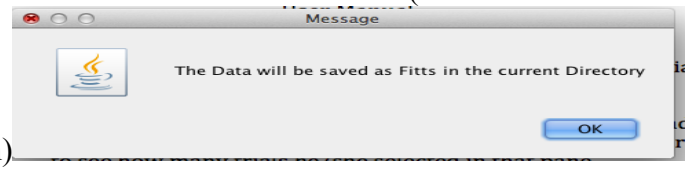


   a. The player is asked if they would like to play
   b. The player can also use the slider to select the number of trials. Trials are between 50 and 100
   c. The no button exits the application
   d. The yes button accepts the trials as entered on the slider and moves to the next box, this clears the Setting Panel box, but the player is till able to see how many trials he/she selected in that pane
   e. Pressing the "x" button will also end the application

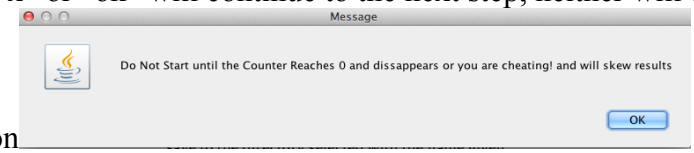
2. Second Step (Specify a file to save)



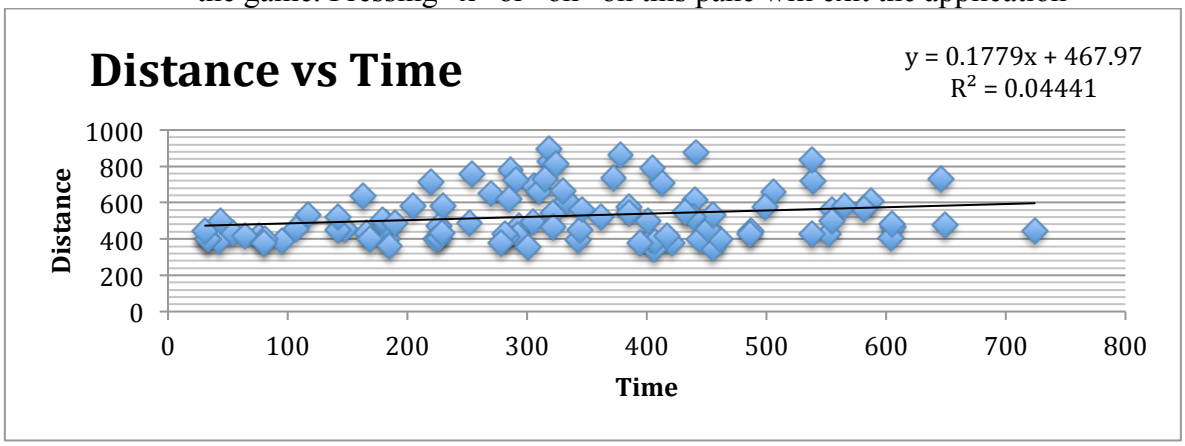
   a. The player will select where to save the output file and what it will be named
   b. It is set to automatically filter to only see CSV files
   c. Choose a directory and name the file. Then press save and the file will save to the directory selected with the name given
   d. If the player presses cancel, exits, or fails to create a file, the file will automatically save as "Fitts" in the present directory. The user will also be prompted when the application takes this action. If prompted, the player must press "ok" or the "x" button to continue. (the "x" button will not end the application)

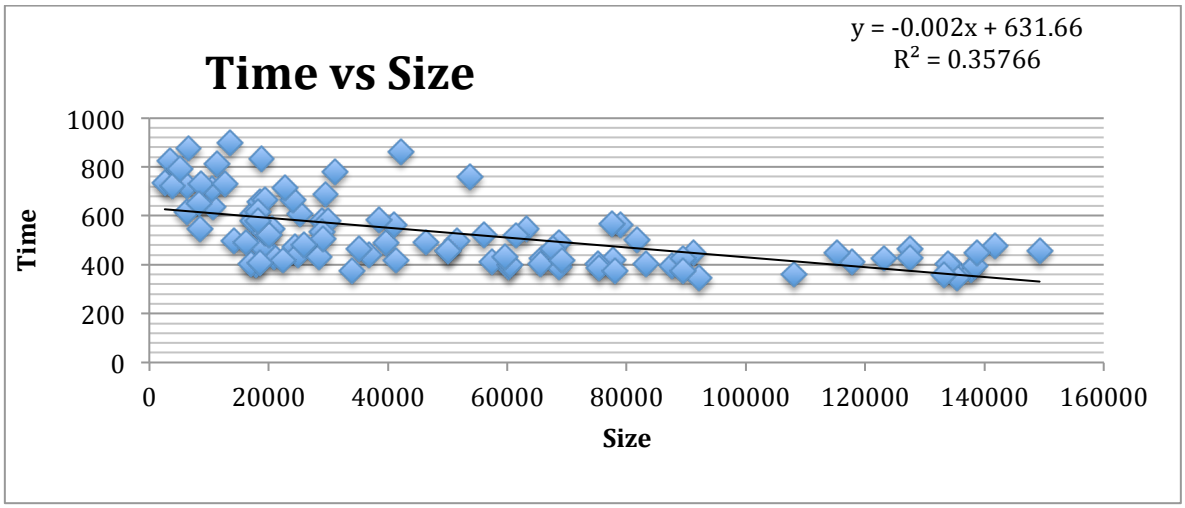


3. Third Step (Message Box: cheater note)
   a. The player is informed to not start playing the game until the timer has completed and disappeared
   b. If the player does not wait for the timer to complete, they are cheating
   c. Cheating will sway the results.
   d. Pressing "x" or "ok" will continue to the next step, neither will end the application



4. Fourth Step
    a. Play the game
    b. After the countdown has completed and the countdown windows has closed, the player may begin the game. Pressing "x" on the countdown window will only hide the box, the game will not end.
    c. The player will click on the shapes until the number of trials set in the slider is reached
5. Game completion
    a. When the player clicks on the last shape, they are notified they have won the game. Pressing "x" or "ok" on this pane will exit the application





Distance did not seem to have a correlation with time.
Size seems to have an ever so slight correlation with the size of an object.
I believe the sample size of 1 person, 1 test, and 100 points, is too slim to really draw any conclusions on. I am also a gamer, so I probably skewed the results with my superior accuracy and reaction time from many years spent in the trenches of WWII fps games. (I'm looking at you Call of Duty…)
I was also using a razer gaming mouse and an Asus gaming monitor (both have minimal lag, especially when compared to the majority of consumer devices).

Appendix A

I made 4 java files.The first Java file, P3.java contains the main method. As instructed, it drives the application. From there, flow of control passes to the FittsLaw.java. The FittsLaw.java file contains the listeners for the buttons in the frame implemented in the P3.java file. After these buttons are pressed, the game either ends or begins. Inside the same file is the listener for the slider.

Fitts.java contains most of the interaction between the 4 files.
I did this because, it made it easier to pass information between the files and contained classes. I would have had less freedom in the static main method. Transferring quickly allowed me to have more dynamic code.

In the FittsLaw.java file, I have the FittsLaw class and the ButtonListener class. These classes are able to take the appropriate actions when listening in on the slider and buttons. From there, flow of control clearly moves downward toward the StartTheGame method.

In the start the game method, flow of control also clearly moves downward. Too keep this donward flow, and to reduce the size of the file, put the countdown in its own java file. That way, it would be easier to debug both files and I would not need to constantly look through large loops. At the end of the FittsLaw.java file, Flow of control moves to the MyCanvas.java file.

This file is where the game takes place. It was much easier to separate out the methods called upon and test them out with their own independent(temporary) main methods when seperated as such.

Each of these file I wrote and compiled indepenedently, before joining them in the FittsLaw.java file.