**CART253 – Creative Computation I – Fall 2021**Pippin Barr

**Progress Report for Project 02**Rachel Boudreau Richard

// WHAT WAS DONE

Since the proposal many advancements were done, or at least it feels like it. The menu prototype has reached its final state in regards of code structure. Coins are looking good, system to link them to their associated games is looking good, intro state is set up and working, and most important of all – I am confident and getting a hang of everything. Classes, states, and how p5 works in general are really starting to sink in so ideas and concepts are becoming much easier to execute.

// WHAT REMAINS

Theoretically, what I accomplished in the past days is what the prototype should have been but hey, we ain’t perfect and late is better than never. I was a little lost and felt defeated on what my project would be but now that it is coming along, I am excited to add and refine everything with my personal touch and get creative with the design. That said, designing the text animation for the intro, designing the coin slot simulation background, adding sounds, refining the games and loading them onto the coins are the things left to do. Hopefully I will be able to accomplish everything with my idea of phone compatibility.

//DECISIONS I HAVE MADE

Chart

Description automatically generated with low confidenceWhen designing the menu, a big challenge was figuring out how to display the coins closest to the vision I had in mind. I started out with images however the transparency of pngs didn’t really translate well in the p5 3D world (see below image on the left). I then tried to create the coins from scratch but the 2D p5 draw world did not enjoy the 3D rotation world (see below image on the right). With Pippin’s help, we considered cylinders, svg files, but him exposing me to textures gave me the thought of using the a texture feature on an ellipse. This was the most successful and turned out to be the final concept used.

A picture containing text, clipart

Description automatically generated