Reflective Paper

Starting this class, I only knew the basics of p5 JavaScript as well as what was taught in 253. I've always struggled with coding, even before this class. My major struggle has always been translating my ambitious ideas into actual coding projects. It's super aggravating to have these concepts, and yet, when it comes to coding time, the simplest things can make the whole project unplayable. Whether it's a game or a simple simulation, nothing seems to go my way. What I've learned through coding is the ability to adapt and improvise—to make something out of something else.

The first thing we did in 263 was create a sound engine, which I found super interesting since it could be very useful in the right hands. One of the projects I did was a voice command game that lets the player control a small square to move from point A to point B. Using their voice, the player must guide the little square from the bottom left corner to the top right corner. Sounds simple, right? The catch is that there are other squares falling from the top to bottom, destroying everything in their path, which means if your square touches them, it returns to the starting point. Using specific voice commands such as "run," "jump," "halt," "go," "walk," and many more, makes the cube feel more alive and ever-learning. The game does sometimes not work, mistaking your words for others. This project made me realize how much more interesting this class will be.

But the project that really shook me was the AI Jam. This project uses a library that allows you to integrate AI into your program. It could turn into a super interactive life simulator, but it was also very complicated for me. What I decided to do was to combine almost everything into one program. As I progressed with the project, I conceived the idea of creating a cybersecurity check. In my program, the user would have to use their camera, microphone, and keyboard to fully experience it. First, the computer would use your camera to identify a face. Once a face is detected, the computer will ask you to place your hand in view for a scan, and so on. This was probably my favorite project to work on because it felt like an advanced "login" page.

The third programming project required creating anything using the Phaser library. Phaser 3 is a popular framework for creating 2D games that run in a web browser. What makes me like Phaser so much is what it offers. Key features of Phaser include support for sprites, physics, animations, audio, and asset loading, which make it suitable for a wide range of games. When looking through all the examples, it seems like the possibilities are endless. The one thing that really caught my attention was a top-down view example. The example showcases a playable soldier and the user controls his aim with the mouse while using a top-down view. This seems simple, but I have always been drawn to games with that camera style, like the old Zelda games on the Nintendo DS. So for the project, I decided to take inspiration from the example and use it for my Prototype project. Initially, I was going to create a simple shooting game but decided to make something more meaningful. So I made a game about cleaning the ocean. The player must navigate through obstacles in the ocean to find and eliminate all the trash (plastic bottles). By adding fishes, it becomes harder to find the bottles. I also added a generic bad guy to shoot at the player for a bit of spice. Although I was not impressed with the final touches, I did have a lot of fun creating sprites, especially the ones with animations. Making the water animation blend in with the surroundings makes the game feel much more alive. Creating objects that fit with the world around them was my favorite part.

For the final project, I decided to elaborate on my prototype project. Using the idea of saving the planet, I thought, why not turn the whole game into an educational experience? Making an enjoyable game and combining it with facts about ocean pollution seemed like a very interesting idea. There were a lot of difficulties in making each scene. Every little gameplay feature I wanted to implement was either met with errors or simply crashed the game. After hours or even days of trying to implement these features, I decided to stick with simpler gameplay. Each scene comes with an explanation and context to help guide the player. One of the things I liked the most from my final project was the sprites in the last scene. I created a collectible item using PiskelApp that blends into the background, making it seem as if it is floating on the water and going in and out of view. The animation made the gameplay more interesting by challenging the player to track the collectible item as it disappears into the water. The second sprite I created for the same scene was a shark that follows the player around, eventually getting too close and "eating" the player if they take too long to evade.

Overall, this class was very enjoyable, especially learning all these new libraries. I would love to use Phaser to make a store game using a top-down view like Pokémon or Zelda. Perhaps even adding voice commands to interact with the NPCs. I know it's way too ambitious, but just the idea of creating new sprites gets me motivated.