# Reflection:

## Knowledge and Skills I learned:

I learned about many intricacies about java as a whole. Just by interacting with java fxgl, I found out that there are many diverse applications of java. From connecting frontend and backend together to generating random graphics and levels, java is an extremely versatile tool for object oriented programming. I also learned the power of just importing packages and tools in, as using their methods instead, as that saves tons of time and effort. Midway through my program, I realized that Larry had imported a collision package that literally could detect collisions with just one method, compared to my way of manually checking each frame 60 times a second to see if the coordinates of the platform will intersect with the player. But I was too far in so I had to finish up with what I had.

Through this experience I learned how to create projects and aim for functionality. Instead of focusing on all the useless things, learning about the order of just making things work before adding extensions of the game really helped me focus on what to get done. I don't know if it's just because I was working alone, but the Kanban board didn't really help me because I already knew what I had to do next. Agile didn't really teach me anything other than crunching. Unfortunately this project didn't really teach me much about time management.

I also learned about the difficulties about project management, as creating a project like a game has many steps and connections that I simply didn't think about before starting. Many key topics like object collision and camera positioning just didn't occur to me, which made this project so much harder.

I didn't use sprites or audio files, so I learned nothing about those parts of creating a game. I did want to try and use them though, I just put functionality over those luxuries.

This project made me realize that I should never take anything programming-related as granted, as there are always hidden steps that show up out of nowhere. Having good organizational skills would help a lot, as well as having another person to work with. But I enjoyed the experience and I just wish I created a better game at the end.

## Problems:

When I started off with libgdx, I quickly realized that the game created in that engine would make no sense to me whatsoever. Simply setting up a camera for the player and window for the game was already getting into complex vectors and such, so I swapped to fxgl instead.

After swapping to fxgl, I realized that I was almost just as lost as I was with libgdx. So I started watching tutorials and reading documentation to get an idea of what to do. After doing all that, I could finally begin on the program. However, I had to spend

3 hours setting up fxgl because I forgot to add the custom build path for the java fx libraries in Eclipse, so the fxgl stuff wouldn't run. I got very frustrated and annoyed during that, but I finally managed to identify the issue after all that time. The github for fxgl installation never mentioned anything about having to install java fx too, and linking the java build path to its libraries from Eclipse either. However, after doing that I could begin coding. So after following a few tutorials and reading some documentation, I finally got the game to function properly and it was alright from there. I did have a few issues with time management though, because the final week had a few math tests as well as summatives from all my courses due at around the same time.

**Survey Password:** "Food falling through the gutters are grate."

Kanban Board: Kanban for ICS Project (with edit perms so you can see the edit history)
Worklog: Devlog for ICS Culminating - Kevin Yao