

MATES Computer Science

**Senior Capstone Project Bi-Weekly Progress Report**

| Project Title | Sweet Tooth |
| --- | --- |
| Team Members | Tyler Depa, Brodie Best |
| Dates Covered by Report | February 15th - March 3rd |

# **Summary of Project** (Provide a one paragraph summary of your project. You can largely copy/paste this from one progress report to the next, unless there are significant changes.)

We are making an action eating game based on a loose concept of Candyland. Currently the idea is that the main character, Plum, will travel the world eating all of her enemies and unlocking different abilities and tools along the way. The plan is to have a simple enemy ai with random enemy generation, 4 procedurally generated environments, and a unique eating system.

# **Summary of Progress this Period** (Provide a high-level, one paragraph overview of what was accomplished this progress period collectively by the team.)

* Set up the GitHub repository and Unity project
* Learned Blender and Unity (C#)
* Created models: Crazyorn, Gummybear, Minty Fowl, Peanut Butter Toad
* Created bones and color for all models
* Began creating a basic system for enemies: Patrolling, Chasing, and Attacking the player.
* Created a player controller, with sprinting and crouching implemented (FOV changes)
* Created an ability system, and made a basic Candy Rush ability (higher speed and jump height, higher FOV)
* Created a basic testing environment, and a working door that opens with the push of a button

# **Detailed Progress this Period, separated by Team Member** (Provide detailed information on the progress that you made in the reporting weeks. Include screenshots of code, your game or website, etc. Each team member should have a separate subsection covering their accomplishments. Not including screenshots, this section should be 1-2 pages.)

Brodie - I created models for the most basic enemies in the four environments that we plan on having. The models were created in Blender using a mirror model and subsequently colored based on the polygons that were used. I then added bones to each of the models, also in Blender, and placed the bones in the rough approximation of where joints would be on the models. I then attached the bones to the models in Blender and added all of them to a file in the GitHub where they can be uploaded onto the Unity project. I tested out uploading some of the models to unity (more specifically the Crazycorn). When I uploaded these models, they were relatively large compared to the environment so they may need to be scaled down. I also began working on the hand models and enemy A.I.

The enemy A.I. will be in a basic 3 parts and subject to change based on the enemy. The most basic 3 parts are going to be Chasing, Attacking, and Patrolling (Patrolling will be done last because chasing and attacking are the most important). The chasing code is a basic setting that makes the enemy position travel toward the Player's position. The enemy will have a sightRange that once walked into will cause it to chase the player. The attacking code will be launching either a projectile or close proximity hitbox when in an attackRange of the player. The patrolling, which will be done later, will have the enemy walking to random positions within set distances and code to stop them from walking through the environment. \*Still currently working on this in the Enemy script.

Tyler - Since the beginning of the semester, we have actually completed a significant amount of work. We started off with setting up the GitHub repository, with folder structure and organization. Brodie and I spent a while planning out the game, where we mostly worked on planning out the lore and some mechanics, and I formatted and set up the readme.md file. We also learned how to utilize GitHub Desktop and were successfully able to transfer the repository across devices. Next, we worked collecting many resources and tutorials. I created a YouTube playlist full of tutorials and tips for Unity that will be useful as the project progresses. At the same time, we did some research and picked our Unity version, 2021.3.18f1 LTS (Long Term Support). We installed WebGL support, which allows for building/exporting the game to be played inside of the browser. This means we can upload the game to itch.io as an executable to download and in the browser with no download. Then I set up the project with some basic settings. As Brodie was learning how to use Blender and started to make models (in detail above), I started on the fundamentals of our game in Unity.

I started off with creating the Player GameObject, a rounded cylinder, and attached the main camera. I followed a couple of tutorials and was able to code a movement and rotation script in playerMotor.cs, allowing control of the Player to move, look around, and jump. Allowing the player to look around was done in the playerLook.cs script, where I controlled the main camera's rotation using the delta x and y inputs multiplied by sensitivity variables (for tuning later). The camera is a child of the Player, meaning all movement of the Player is also applied to the camera. As far as movement, this was done using Unity’s new input system, which easily allows adding different inputs (like keys on the keyboard) to be used in scripts. Originally, the script just detected if a key was pressed (like w) and changed the Player's position every frame. It also detected the crouch and sprint keys, and toggled the action. These increase/decrease the Player’s speed, crouching changes the Player’s height and jump height, and sprinting changes the main camera’s FOV for a nice effect. I also learned lerping (linear interpolation) to smooth the fov and height transitions. We then took a week or two updating software, planning the game, and creating the presentation pitch, project scope, and other documents. After this, I refined the movement script, changing the speed and jump height values to be multipliers to the walking speed instead of set values (for easier modification later). I also made these actions trigger on hold of a key instead of toggling on and off on each press.

Next I started working on the abilities. I created a script that handles all of the future abilities (player ability.cs), and created a basic ability called Sugar Rush to test it out. I was able to set a duration for the ability to last, and set the players FOV, speed, and jump height to be faster in that time frame, along with tinting the screen pink (harder than you think). At this point, I also expanded the testing environment, adding many boxes and slopes to jump around on to text the movement controller. I was also able to code a simple interactable script that will handle different interactable objects in the future. I was able to make a raycast shooting out a certain distance from the center of the Player’s camera, detecting when it passed through an interactable. So far, I created a button that when hovered over and hit E on the keyboard, opens a door next to it (I also learned simple Unity animations).

# **Difficulties Encountered this Progress Period** (Provide detailed information on the difficulties and issues that you encountered in the reporting weeks. Discuss mitigation strategies for how you got around or plan to get around these issues.)

Tyler: Mostly just little bugs I encountered while scripting since I started off not knowing the language too well. There were also some bugs that we sorted out relating to GitHub and sharing the project across computers. Most of the Unity work had to be done at home due to the 2nd block class not getting software installed until this week.

Brodie: I had a hard time working with the bones in Blender in the beginning but I found some easy fixes online and was able to solve the problem pretty fast. I am still trying to get the hang of animation in blender (I may use Unity) so we'll just have the models frozen in-game for now.

# **Updated Gantt Chart and Discussion** (Provide screenshot of and link to updated Gantt Chart. Discuss any changes made to the chart since the last progress report and why.)

<https://docs.google.com/spreadsheets/d/1szyNSMEZ3MuRSzZiWZNMSg-Nt1oocHF3aQVfg9pDXwE/edit?usp=sharing>

Brodie: I’ve decided that it is easier to make models, bones, and design all at the same time for the enemies so I am going to have to edit the Gantt chart to accommodate all of this. It is actually a lot easier to make certain models than I thought it would be so this task may be expedited.

# **Tasks to Be Worked on in Next Progress Period** (Discuss the tasks to be worked on in the following two weeks. Discuss who is working on each.)

Brodie: Working on more models, adding bones, learning Blender/Unity animations, importing the models into Unity, creating enemy AIs

Tyler: Working on tool scripts, post processing, procedural generation/biomes, projectile physics

# **Additional Information** (Provide any additional information that you want to provide in this section; for example, one of your teammates is going away next week, your Github account is gone, etc. It could be good news as well.)