Will Harris

Sammy Gaskin

Evaluation Document

With this game, our intent was to create a game that would encapsulate some of the difficulties involved with being as slow as a snail. The defined narrative of the game is that the snail must mail some letters, but he doesn’t have any stamps and he doesn’t know where the post-office is. So, the challenge is to steal as many stamps as possible while avoiding guards and finding the location of the post office. If the snail is caught by a guard, the stamps he collected are worthless since he can’t mail his letters in snail prison.

The main mechanic of the game is the movement of the snail. The snail is able to move using the ordinary arrow or wasd keys. The snail is able to interact with objects by moving into them, which makes it easy to interact with the game world. He can pick up stamps, read signs, and go into the post office in this way. The guards follow a consistent pattern of movement where they circle a given stamp. Every stamp is spawned with its own guard and the guards don’t go away when the stamp is collected. This causes the number of guards to increase as the game goes on, making it harder to collect a large number of steps.

We chose a city tilemap for our game because it feels like it would be easy to make a post office in the game world, plus the tilemap has grass tiles to create wide areas for the player to move around in. The tilemap also has a wide variety of building tiles, allowing the buildings to be varied enough to avoid confusion. The tilemap has a sixteen-bit look that helps in making the game look nice and less like a couple art pieces put together. The rest of the sprites used don’t share that same look, but they still are rather nice. We found a snail sprite on Open game Arts by Flatlander, and we chose it because it is a very simple sprite sheet, it looks appealing, and if we wanted to animate the snail moving it wouldn’t be hard. Though, we did not end up animating a walk cycle for the snail. The guards we also found on Open Game Arts. The specific one we used is from megupets, and we chose it because the sprite has a very quirky look to it. It is almost comical with the hair being almost bigger than the head, so it fits with the non-serious tone we sort of made for the game. The stamp sprites we used were the only ones on Open Game Art that looked somewhat like stamps. But, they still are rather nice. The stamps are from Nia Mi, and they add some variety to the aesthetic. Consequently, it makes the stamps easier to notice in game. The last sprite we used is a sign from Nemisys on Open game Arts. It fits in with the sixteen-bit aesthetic of the world, so the game feels more cohesive.

Joys and Struggles:

Sammy

In making this game, I had some difficulties figuring out how everything was laid out in 2d. For example, I had objects rendering in the wrong places like the snail trail showing up behind the street, or the sign colliders not seeing the player. Scenes also took some time to figure out, especially figuring out how to make parts of the UI show up in some scenes but not others. I also ran into difficulties with Github when I worked on something at the same time Will do. Gitkraken could handle merges fairly well, but changes to the tilemap caused merge conflicts.

I enjoyed seeing everything in the game come together. The UI fit together fairly well despite some difficulties with fitting everything on one persistent canvas. I also thought the different aesthetics of the stamps turned out fairly well, with them spawning particles and a sound effect when collected.

I worked on the scene changes, set up the initial movement for the guards, set up stamp spawning, and helped connect all the pieces of the stamp pick-up events. I also worked on bug fixes, such as fixing the problems I mentioned earlier with the snail trail and sign colliders.

Will

Some of my difficulties in making the game is to have mechanics work the way we specifically wanted to, as well as creating a game world that works with and not against the mechanics. One of these is the camera, which we had some difficulty with. I had difficulty when I inserted lerping and then set the boundary percent to 35. The camera did not like that and I spent some time trying to figure it out when I set the percent to 20 and then the camera worked smoothly. Another issue was in getting the colliders and the triggers to work properly. Sometimes, the issue was in how I wrote the code. Other times, it was because I set the wrong collider on the object. But, the colliders work well now.

The greatest joys I had was in creating the game world, adding details to make it feel like more than a flat plane, and messing with the particle system. That is mostly because the hard stuff - movement, camera, and core gameplay loop - was all behind me. Plus, Unity’s particle system is fairly intuitive and fun to mess around in.

I worked on creating the movement for the character, setting up dialog boxes, creating the game world, setting up tilemaps, setting up colliders to act as borders and triggers, and creating the pick-up particles for the stamps. I also helped out a little in some bug fixes, making things flow more smoothly, or just doing some minor adjustments to the game a little bit more fun to play, like making the guards circle around the stamps in the patrol script instead of being locked to the x-axis.

Attributions

* Chime sound recorded by KevinGC downloaded at: <http://soundbible.com/1598-Electronic-Chime.html>
* Sign Post by Nemisys downloaded at: <https://opengameart.org/content/lpc-sign-post>
* Medieval wax stamp by Nia Mi downloaded at: <https://opengameart.org/content/medieval-wax-stamp-letters>
* Sassy Guard by megupets downloaded at: <https://opengameart.org/content/sassy-guards>
* Snail by Flatlander downloaded at: <https://opengameart.org/content/snail>
* Modern City Top-Down Tileset by Emily2 downloaded at: <https://emily2.itch.io/modern-city-top-down-tileset-16x16>