CS 135 Final Report – Peaceful Gardening

Tasmiyah Qazi

James Bartulis

Abstract:

Our project is a peaceful gardening simulator for the Oculus Rift. The objective of the game is to grow and harvest as many crops as you can. It allows the user to cultivate and grow 4 types of crops (pumpkins, turnips, eggplants and tomatoes), using a hoe, a watering can and a time skip mechanism. The features used are: trigger zones for the hoeing and watering interactions, a fade in and fade out mechanism for the time skip and general body mechanics to be able to grab items and harvest crops with the Oculus Integration. What makes our game unique is that it's designed to be as peaceful and easy as possible. It features an in game easy to follow tutorial, and a relaxing background noise of chirping birds. The only other sounds used are for the hoeing and water interactions.

Note: Our project was developed on Unity version 2018.3.0f2

Introduction:

Peaceful Gardening is a gardening simulator made for those who don't have a garden, who would like to relax and de-stress, who would like to learn or for entertainment. The game starts out by the user fading into the middle of a garden and immediately greeted with an in-game tutorial which is on a sign in front of them. The user then can choose which crops to grow by reading the signs and choosing which patches of soil to cultivate. They will be able to move themselves and the camera with the oculus controllers (default settings of Oculus Integration Asset). To be able to grow an instance of a crop fully, the user must till the patch of soil dedicated to that plant and water it 3 times, using time skips in between to wait for the plant to grow. When the plant becomes a fully-grown crop, the user can then grab it and place it in one of the carts placed on the scene to harvest it. To make the game frustration free and so that no one must bend-down, everything is placed waist-level. If an object falls to the floor, it will always respawn from where you picked it up. We've also made the time skip transition a slow one (when the user presses 'x'). The game also features a total of 3 sounds to not overwhelm the user. A constant background noise of softly chirping birds, the hoe interaction noise and the watering noise.

Some other reasons why this project is interesting is that since it is targeted to be peaceful, it is extremely simple. The user is not bombarded with too many tasks or interactions. They can play at any pace. There is no aspect of challenge either. For example: the plants will not die if you don't water them. Also, one of the Detailed Albedo materials used for soil is created by us using PyxelEdit.

The VR experience really makes you feel like you own a garden. One that is easy to take care of. Simple but still immersive. The birds chirping in the background make you feel like you

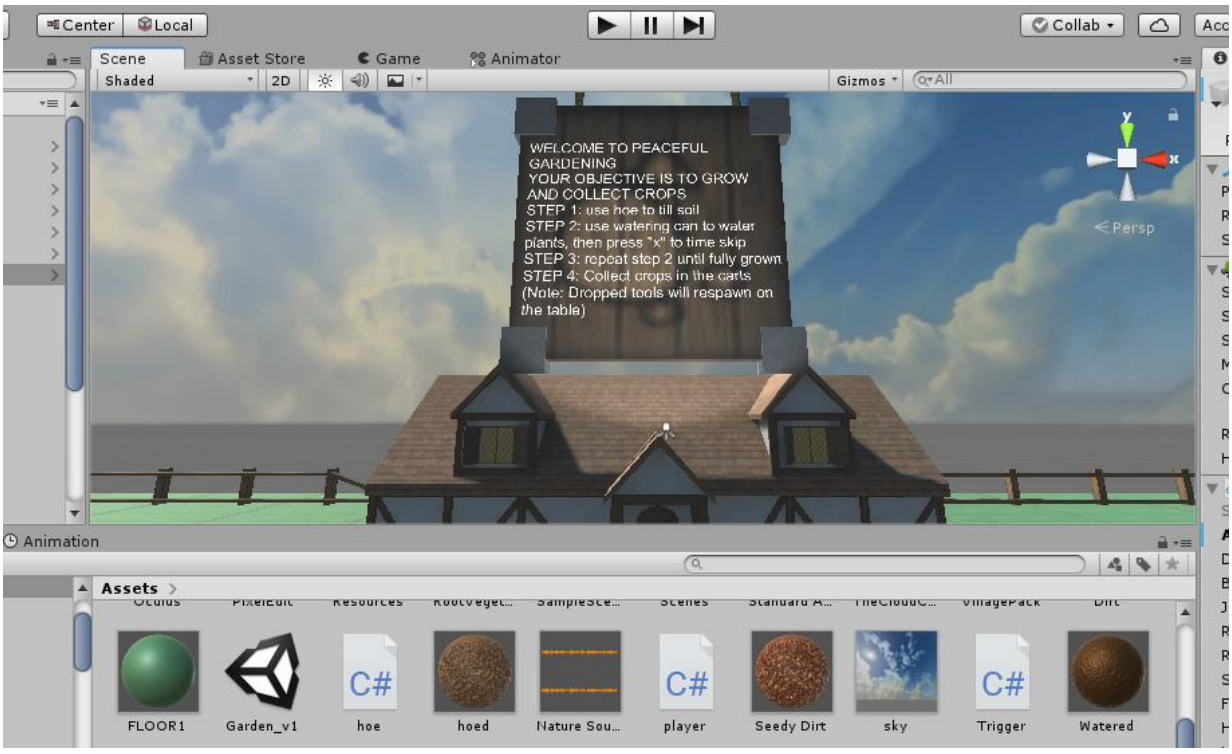
teleported to a garden and the object interactions are still natural. For example, you still have to tilt the watering can to water the plant, and you still have to drag the hoe on the soil to till it.

Related Works:

We were only able to find one other game called Potioneer which was a similar gardening simulator. Potioneer is a more in-depth simulator which also includes interactions of seeds, adding more dirt with a shovel and real-time plant growth. Potioneer also features a ray cast from the controller to teleport to where you want to go rather than moving with the joystick. In comparison, our project was simpler in terms of interactions, we did not have to add dirt, plant seeds or re-fill the watering can. Our project also had much less to render. But Potioneer was different ways where they didn't till their soil and couldn't walk around using the joystick. Their instructions/tutorial was also hard to find. Our project also included a realistic time skip rather than real-time growth. Potioneer also didn't have crops or plant interaction. One thing I think is big disadvantage is that in Potioneer everything was done on the ground. This made the user visibly fatigued. Peaceful Gardening doesn't allow any ground interaction.

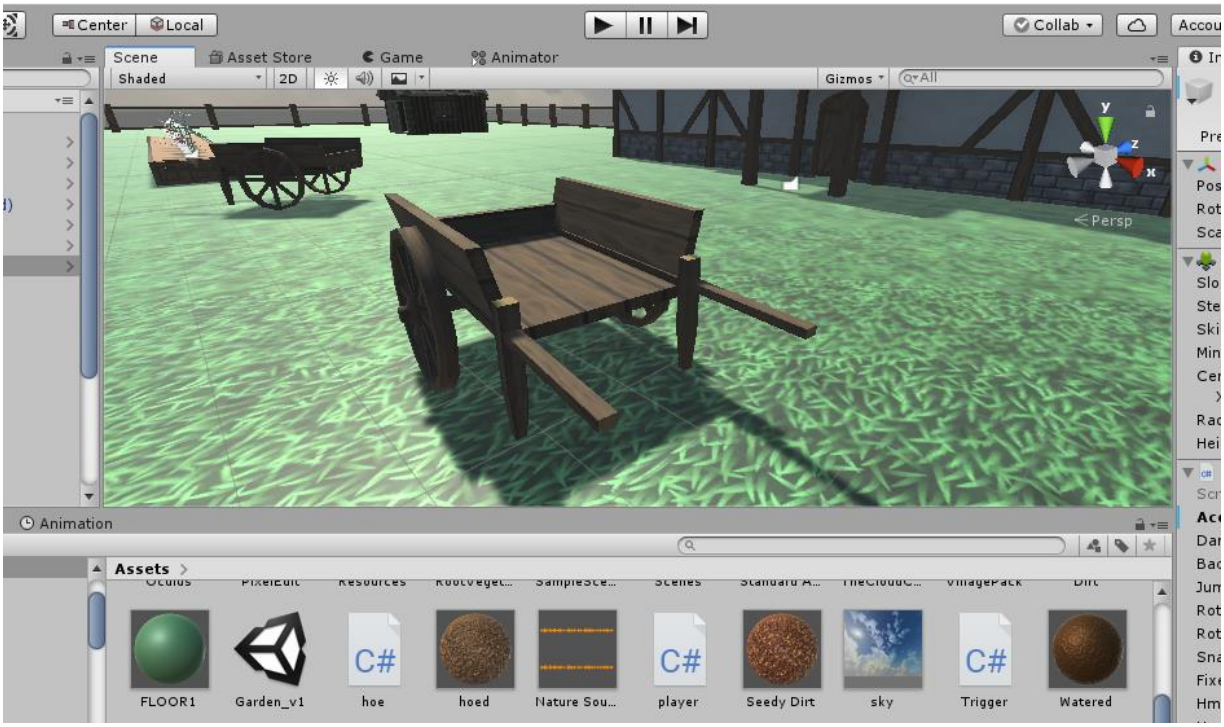
Design:

What the user sees upon entering the game:

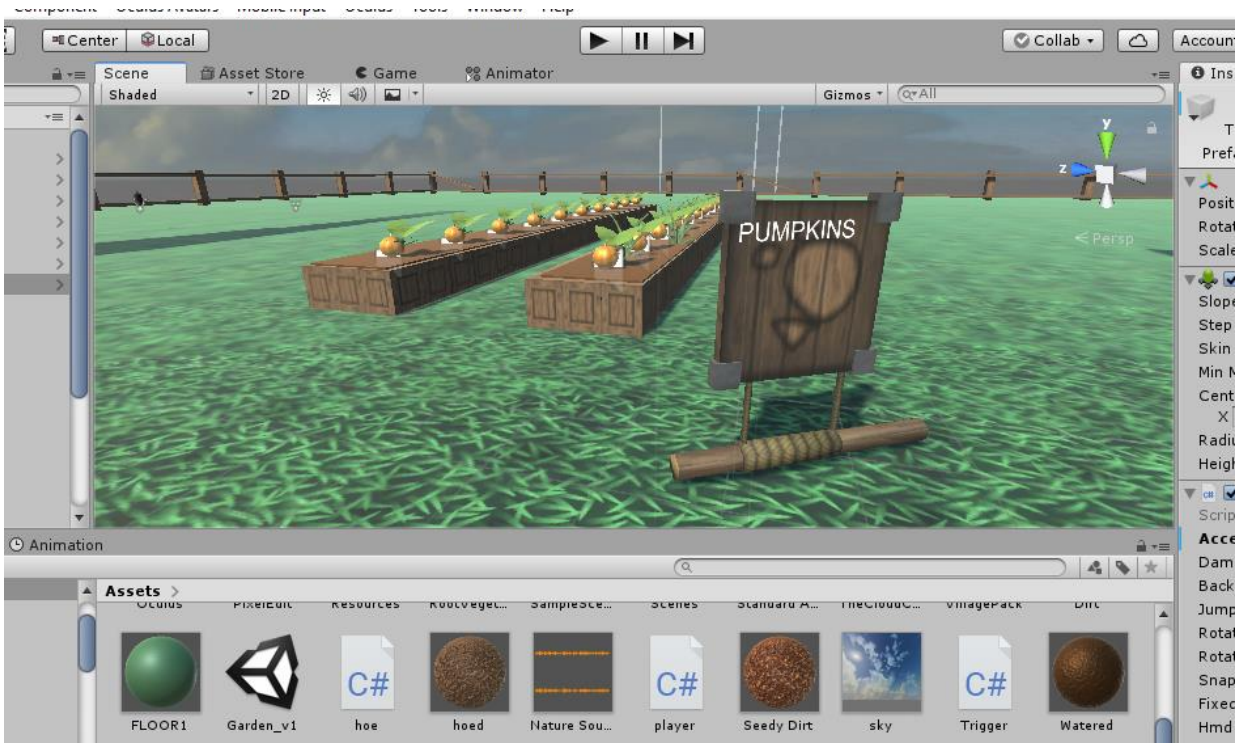


The tools you can interact with (hoe and watering can):

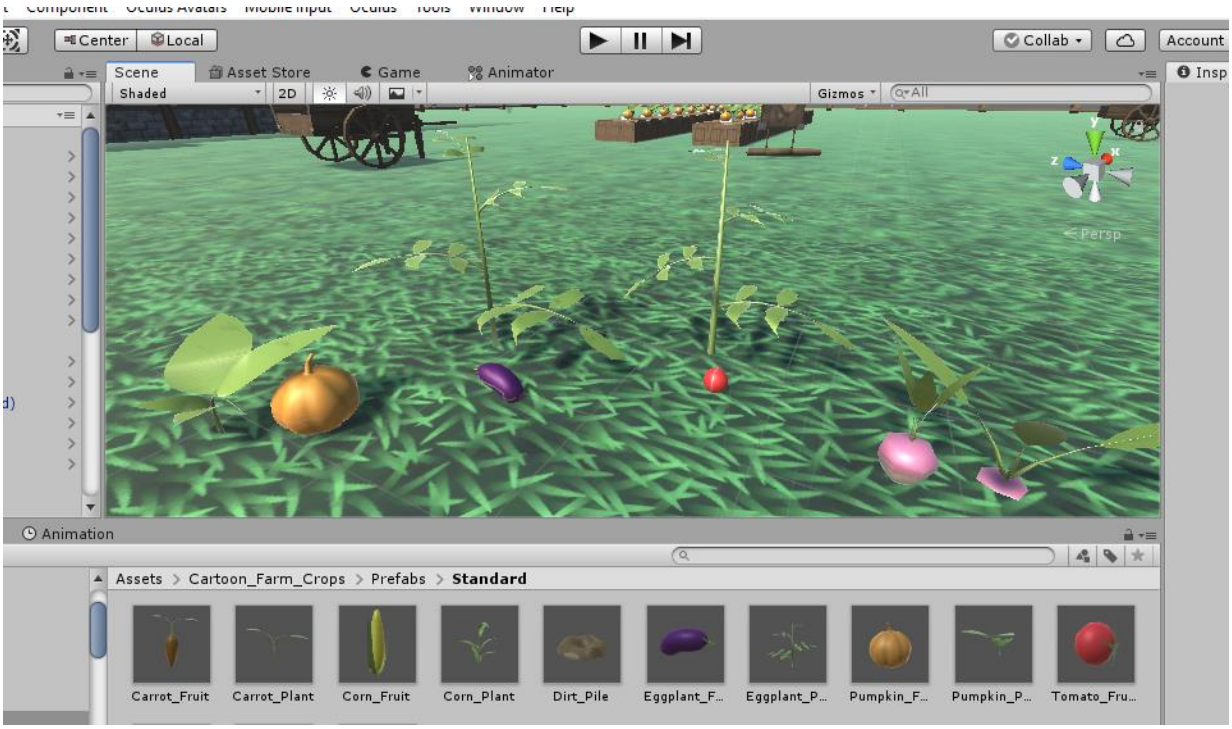
One of the carts to collect your crops in:



One of the four soil patches you can grow crops in (this is the pumpkin patch):



Examples of the 4 crops along with their plants (In the game there are 3 plant growth stages):



We chose only the Oculus controllers as our only means of user interface. For a VR game it is important to make the experience as natural and integrated as possible. So, the user can move themselves and the camera using controller joysticks, grab objects with the oculus triggers and lastly press ‘x’ to time skip. The controls to actions in real world are mapped to exactly how the action would be carried out in real life. For the hoe interaction, you need to touch the soil and drag. For the watering interaction, you need to hover over the sight and tilt. There is sound for each as well.

According to the Oculus best practices guide, we are conforming to almost everything. Everything is slow paced except for the user acceleration. Instructions are clear to the user and we are using default button mapping, a strong sense of avatar and object scale and careful rendering.

A user would best perform in an ideally spaced room with the sensors being able to see the user’s whole body. Otherwise it can be a little difficult leaning over the table in VR to pick up an object. Some users experienced motion sickness when moving in the virtual world. As a result, we should slow down the oculus player acceleration next time.

Some other design decisions we made were our choice of assets. We didn’t want a low resolution feel to our game or too much of a fantasy feel either. Our goal was to make it look realistic enough to feel like you are in an actual garden which can be replicated in real life.

Implementation:

We used 4 scripts in total. The hoe and watering can scripts contain the audio sources for each interaction as well as the feature of respawning the item if it touches the floor. The script connected to the player governs the fade in and fade out animation, exiting the game and player on collision detection to make sure the object the user is holding doesn’t push the player. The last script is connected to each soil instance and governs which soil material to render and which plant to render according to the stages of the plant cycle. This script also is in control of the trigger zones for the watering and hoeing interactions and playing the sounds.

The list of assets we use are:

- Cartoon Farm Crops
- Classic Skybox
- Footstep (snow and grass)
- Footstep SFX pack 1
- Garden Tools
- Medieval Town Exteriors
- Nature Texture Pack
- Oculus Integration
- Standard Assets
- Toy Sprinkling Can
- Village Pack

We also converted a youtube video to wav format for the background sounds of nature and chirping birds: https://www.youtube.com/watch?v=DqewBvd-bAA

For one of the soil materials we created a detailed albedo using PyxelEdit. PyxelEdit was used while developing to try out some custom-made textures. I exported the images created in pyxel edit to the address folder where I changed its settings once in unity to be a sprite. And applied it in various combinations to new materials and existing ones until a favorable outcome was produced.

In terms of challenges or issues during implementation:

- The watering can sound would not play in the OnTriggerStay function. After research and debugging, we found that moving it to play in OnTriggerEnter at least made sounds upon enter but not when the watering can was inside the trigger zone
- When the user grabbed an object and it collided with the user avatar, the user would get pushed back. To fix this we used OnCollisionEnter on the player and used physics.Ignore when an object with a particular tag collides with the player.

Other than that, we did not use any code from the internet other than help from Unity Forums if needed.

Lessons Learned:

Some lessons we learned along the way is that game development is a tedious process and a lot of time and effort is needed for desired outcomes and a tidy feel for the game. We also learned how to use PyxelEdit and learned a completely new concept in Unity which is creating animations.

During demo some users complained about the speed of the player. This can be improved by reducing the acceleration. I would also like to add in a head bob for natural walking feel.

Another complain was that they were getting pushed back from holding an object. This issue was soon fixed.

I also noticed people having a hard time finding the tool table. We can improve this by adding a sign.

Workload:

Tasmiyah:

- Asset and sound research
- Hoe and watering interactions
- Scene set-up and creation
- Trigger scripts for soil
- Player set-up and avatar holding interactions
- Player script

James:

- PyxelEdit material creation
- Fade in/out animations
- Tutorial and sign creation
- Fixed collision bug