

# MULTIFARIOUS



VGD Game Design  
Bachelor Thesis, ZHdK June 2023  
Luca Imesch & Anja Irniger

# IMPRESSUM

Bachelor thesis | ZHdK - Zürcher Hochschule der Künste

Specialization | VGD Game Design

Candidates | Luca Imesch & Anja Irniger

Start of studies | HS 2020

Study duration | 6 Semesters

Mentoring | Stefan Schmidlin & Florian Faller

Zurich | 08.06.2023

## Special thanks to:

*Our Mentors Stefan Schmidlin & Florian Faller,  
to Reina Burkhalter for composing the music and sound  
effects for our Game and to our classmates for their open  
ears, helping hands and inspiring inputs.*

*We, Luca Imesch & Anja Irniger, confirm that the work for this  
paper and the game called "Multifarious" was entirely  
done by ourselves and that no additional help was provided  
from sources other than those allowed.*

*All sections of the paper that use quotes or describe an argu-  
ment developed by an external individual have been referen-  
ced, including all secondary literature used, to to clarify that  
said material has been adopted to support our  
thesis.*

*Luca Imesch, Zurich, 08.06.2023*



*Anja Irniger, Zurich, 08.06.2023*



# TABLE OF CONTENT

1.0 Abstract	1
2.0 Introduction	2
3.0 Research	3 - 15
3.1 Games Research	4 - 9
3.2 Conceptual Research	10 - 11
3.3 Fungi and Mushrooms	12 - 13
3.4 Exobiology	14
3.5 Refined Questions	15
4.0 Process	16 - 36
4.1 Challenges	17
4.2 Art	18 - 29
4.3 Coding	30 - 36
5.0 GDD	37 - 48
5.1 GDD	38 - 48
6.0 Conclusions and Outlook	49 - 54
6.1 Conclusions and Outlook	50 - 54
7.0 Sources	55 - 56
7.1 Sources	56

# 1.0 Abstract

*This paper shows the process that we have gone through, the problems we have faced and the observations we have made during the creation of Multifarious.*

Multifarious is a Game about a mycelium trying to survive and eventually thrive through expanding and entering a symbiosis with the forest around it. The players will each take control of a small mobile part of the same fungus network and try to connect it to various natural resources, expanding the mycelium in the act. Beware of the natural predators, which will periodically find their way to the fungus and try to feed on it if not fended off in time.

*Our leading question represents the core of Multifarious. It has guided us through the entire process and has influenced all of Multifarious's features, from Mechanics to art style.*

"How can the defining features of fungi and mycelial networks be converted into multiplayer co-op gameplay mechanics that incentivise players to regularly communicate and coordinate?"

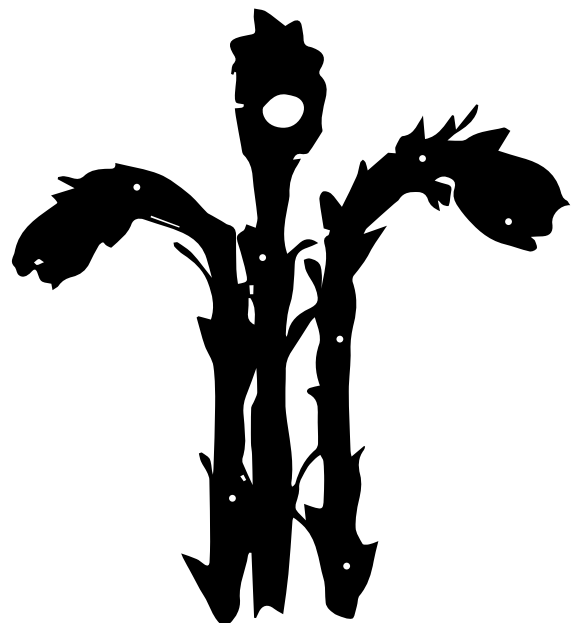


## 2.0 Introduction

*What do we want? Why multiplayer?*

Oftentimes, multiplayer games can and will be played in complete silence, without much need for coordination or cooperation. While this can be a valid type of game, we wanted to create the exact opposite – a game that incentivizes communication between players by creating problems and friction that can only be solved through working together and coordinating.

To determine how to create a multiplayer environment with a continuous discourse, we examined several different multiplayer games.



## 3.0 RESEARCH



# 3.1 Games Research

*In this section, we will analyze various games we have looked at or played to explore how they incentivize communication among players. To obtain precise information from our analysis, we asked the following questions for each game, where applicable:*

*What is the game about?*

*How does the game incentivise communication?*

*How does the game motivate players to specialize?*

*What learnings or questions do we take from this?*

- *Nobody saves the world*

What is the game about?

Nobody Saves the World is a game where the player starts as a "Nobody", someone without any special abilities or skills, and has to save the world. The player unlocks the ability to transform into different forms, which forms the backbone of the game's mechanics. This unique shapeshifting system allows players to unlock new forms by mastering the previous ones through questlines. As players unlock new abilities for the acquired forms, they can then combine them with other forms to discover exciting ability combinations to overcome challenges.

How does the game motivate players to specialize?

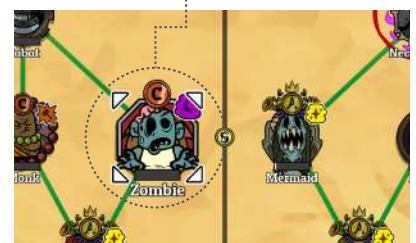
Players have the freedom to individually train their favorite forms and switch between them as they prefer. It is often beneficial for each player to play different forms with distinct strengths, compensating for each other's weaknesses. This approach to specialization allows players to have agency and feel rewarded when thoughtfully selecting and combining their chosen forms.

How does the game incentivize communication?

Most of the combat in this game takes place inside dungeons scattered around the overworld. Each dungeon has specific "skill requirements" in the form of damage type weaknesses that can be overcome by selecting the appropriate shapeshift



[Figure 1: Nobody Saves the World, DrinkBox Studios, 2022]



[Figure 2: Nobody Saves the World, DrinkBox Studios, 2022]

forms. The majority of communication occurs when entering a new dungeon, as players need to adapt and discuss which forms to use in combination to cover all the dungeon's prerequisites.

What learnings or questions do we take from this?

---

A system that allows players to individually train and combine abilities encourages specialization, especially if these decisions can make upcoming dungeons easier for the team. The individual combination of abilities within forms is more important than the overall combination of forms as a team. However, since this discussion mostly happens at the beginning of a new dungeon, there may be limited talking points. To generate more discussion, the game could periodically shift dungeon requirements or introduce additional friction that influences how players choose their forms.

## o *Minecraft*

What is the game about?

Minecraft is a sandbox game that offers an abundance of possibilities, catering to a wide range of player interests. While Minecraft can be played solo, its multiplayer aspect holds tremendous potential. For our analysis, we focused on the scenario of two players starting a Minecraft world together, aiming to achieve their goals while surviving and thriving in this world.

How does the game incentivize communication?

During the early stages of the game, there are numerous interactions as both players explore their surroundings, searching for various things based on their priorities: animals, food, wood, iron, or a suitable place to establish a shelter. This initial exploration prompts extensive communication as players share their findings, locations, and more.



[Figure 3: Minecraft, Microsoft, 2011]



[Figure 4: Minecraft, Microsoft, 2011]



How does the game motivate players to specialize?

---

As the game progresses and things start to settle, players often divide tasks to establish a solid foundation swiftly and prepare for potential dangers, such as nighttime encounters with zombies. While gathering resources and expanding their base, players may plan, assign tasks, and then focus on their respective jobs for a period of time, requiring minimal further communication until the planned tasks are completed, which can take hours depending on the nature of the tasks.

What learnings or questions do we take from this?

---

Joint explorations generate the most communication as both players aim to stay together or at least in close proximity while discovering new things or even searching for specific objects, animals, or structures. The game does not force players to specialize but instead offers a wide variety of playstyles, providing players with the freedom to pursue their individual interests.

---

- *Portal 2*

---

What is the game about?

---

Portal 2, like its prequel, is a puzzle game that utilizes portals and the altered physics they introduce to create a wide variety of puzzle rooms. Players create portals by shooting their portal gun at walls, with only two portals allowed to be active at a time, directly connected to each other. Additionally, there is a multiplayer mode where the second player also possesses a portal gun with its own set of two portals. This addition increases the complexity of the puzzles by introducing more possibilities.



[Figure 5: Portal 2, Valve, 2011]

---

How does the game incentivize communication?

As a puzzle game, the primary topic of discussion revolves around solving the level. However, most of the time, one player immediately sees the solution or discovers it independently, leading to a situation where they simply "command" the

other player on what to do to complete the levels. Thus, much of the communication leans towards giving commands rather than collaborative problem-solving. This asymmetrical gameplay is created by designing problems with only one solution or requiring an understanding of a specific detail in the level.

How does the game motivate players to specialize?

---

This game does not allow players to specialize, as they have identical tools and cannot modify them in any way.

What learnings or questions do we take from this?

---

Creating puzzle games with a single solution per level can make it challenging to foster vibrant discussions during the solving process. This issue can be addressed by either designing multi-staged puzzles where all parts are visible simultaneously, encouraging players to approach the solution from different angles, or by providing each player with different tools, making it more difficult for them to determine what they can do individually and what the other player can do. This approach incentivizes players to uncover their own pieces of the puzzle.

## ◦ *It takes two*

What is the game about?

---

It Takes Two is a co-op action puzzle game where players assume the roles of parents transformed into toys. The game is inherently asymmetrical, with each player controlling a different avatar possessing unique strengths. The mechanics of the game change as the players progress through different stages, often accompanied by new tools to overcome challenges.



[Figure 6: It takes two, Electronic Arts, 2021]

How does the game incentivize communication?

Similar to the previous game mentioned, the primary focus of discussion in It Takes Two revolves around solving puzzles. However, in this game, the emphasis leans more towards joint problem-solving due to its inherent asymmetrical nature.

How does the game motivate players to specialize?

---

Players become specialized in the game through the changing mechanics. As most of these asymmetrical mechanics are tied to specific stages, the game maintains a fresh experience and avoids frustration, despite players being assigned specific roles.

What learnings or questions do we take from this?

---

Although players cannot individually specialize or customize their characters, the game remains engaging through its fast-paced nature and dynamic mechanics.

- *Dome keeper*

What is the game about?

---

Dome Keeper is a game where the player character lands on a foreign planet and strives to establish a foothold there. The starting point and base of operations is a glassy dome, which the player can control to defend themselves and upgrade both the dome and their character in various ways. The game cleverly separates the "outside world" where enemies periodically attack, from the "inside world" where the player ventures underground to gather resources and unlock upgrades.

Dome Keeper is not a multiplayer game, but it serves as a significant inspiration for creating the necessary cooperation and specialization dynamics in a survival scenario, making the game replayable and engaging.

What learnings or questions do we take from this?

---

The delicate balance of venturing underground to gather resources and returning in time to defend the dome creates substantial tension and allows players to optimize their approach. This optimization can be achieved through different upgrade paths. Do you prioritize upgrading resource gathering to be more efficient in the long run? Or do you focus on enhancing defense mechanisms to repel enemies quickly, minimize damage, and



[Figure 7: Dome keeper, Raw Fury, 2022]



[Figure 8: Dome keeper, Raw Fury, 2022]

expedite resource gathering?

These decisions become even more intriguing when considering the addition of a second player as a thought experiment. With two players, one could specialize in continuous resource gathering while the other primarily defends the base. However, what if both players manned the defense dome together and then gathered resources as a team? Would this lead to a different allocation of upgrades based on playstyle? Would individual player upgrades be an interesting addition?



[Figure 9: Dome keeper, Raw Fury, 2022]



## 3.2 Conceptual Research

*Cooperation typically occurs when it becomes necessary. By default, people tend to be lazy and seek the path of least resistance. If a game does not require optimization to complete a given task, there will be no discussion or coordination involved. Therefore, the game needs to push players in the direction of strategizing and specializing their characters for certain tasks.*

*During our research, we discovered various ways to incentivize cooperative coordination, and the most promising ones are documented below.*

- **Possible Solutions for Co-op Motivation:**

### Efficiency & Limitations

Limitations breed creativity. By imposing well-chosen limitations on players, they are compelled to think outside the box and discuss potential solutions with each other in order to succeed.

Efficiency is only achieved when the game is designed to require players to act efficiently. This can be accomplished through external forces or requirements. For example, a time limit may be imposed, necessitating players to plan their actions strategically to meet the deadline and progress in the game. Other threats, such as enemy appearances that can kill characters if not efficiently fended off, can be utilized similarly.

Through these means, players may start splitting up tasks, striving to improve their performance, or setting personal goals, which further enhances their willingness to optimize together.

### Positive Reinforcement

In games with strategic elements, splitting up tasks and specializing are often the most efficient approaches. Positive reinforcement and a sense of being needed are crucial factors to consider in achieving cooperation.

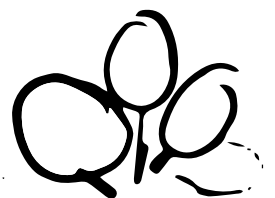
Game mechanics can be employed to overcome certain challenges when players specialize with specific abilities, effectively creating a "rock-paper-scissors" mechanic where each player can overcome certain challenges but not others, or at

least not as efficiently as the specialized player. Additionally, positive reinforcement can be reinforced through sound effects and visual feedback. If an ability feels enjoyable to use, players will be more incentivized to specialize in that role.

### Meaningful And Open Problems Requiring Creative Problem Solving

---

To encourage discussion and strategizing among players, the game should present problems that have multiple potential solutions rather than a single correct answer. While creating such problems can be challenging in puzzle games, it is feasible in other genres like roguelikes. In roguelike/roguelite games, this is often achieved through the different upgrade options available to players. During each run, players can experiment with different upgrade combinations to find a viable solution for completing the game. The openness of the problems provided by the game in co-op roguelike games can spark conversations as players collaborate to find the best solutions when building their characters.



## 3.3 Entangled life by Merlin Sheldrake

*The information gathered from this book was a significant source of inspiration for our project. It contains numerous valuable facts and knowledge about fungi and helped us further develop our game mechanics and art style. The following is a small selection of facts that have been incorporated into our game.*

1  
"This means that the carbon that powers the life of *Monotropa* - the bulk of the stuff from which they are made - must ultimately come from other plants, via a shared mycorrhizal network: if carbon didn't flow from a green plant to *Monotropa* through shared fungal connections, *Monotropa* couldn't survive."

Merlin Sheldrake, *Entangled Life* [2020], S. 166

3  
"Mycorrhizal networks can help to overcome these limitations, in some cases providing a 'fungal fast lane' or 'superhighway' for plants that broadcast poisonous deterrents." "Wood wide webs, in other words, are about far more than the movement of resources - whether energy-rich carbon compounds, nutrients or water. Besides poisons, hormones that regulate plant growth and development can pass through shared mycorrhizal networks. In many species of fungus, DNA-containing nuclei and other genetic elements such as viruses or RNA are free to travel through the mycelium, suggesting that genetic material might pass between plants via a fungal channel - although these possibilities have barely been explored."

Merlin Sheldrake, *Entangled Life* [2020], S. 181

4  
"...fungi can direct the flow by growing, thickening and pruning back parts of the network - or indeed, fusing with another network entirely."

Merlin Sheldrake, *Entangled Life* [2020], S. 178



2  
"For the relationship to thrive, both plant and fungus must make a good metabolic match. It is a familiar pact. In photosynthesis, plants harvest carbon from the atmosphere and forge the energy-rich carbon compounds - sugars and lipids - on which much of the rest of life depends. By growing within plant roots, mycorrhizal fungi acquire privileged access to these sources of energy: they get fed. However, photosynthesis isn't enough to support life. Plants and fungi need more than a source of energy. Water and minerals must be scavenged from the ground - full of textures and micropores, electrically charged cavities and labyrinthine rot-scapes. Fungi are deft rangers in this wilderness and can forage in a way that plants can't. By hosting fungi within their roots, plants gain hugely improved access to these sources of nutrients. They, too, get fed. By partnering, plants gain a prosthetic fungus, and fungi gain a prosthetic plant. Both use the other to extend their reach."

Merlin Sheldrake, *Entangled Life* [2020], S. 139

5

"Fungi supply other crucial nutrients to plants, such as zinc and copper. They also provide plants with water, and help them to survive drought as they've done since the earliest days of life on land."

Merlin Sheldrake, *Entangled Life* [2020], S. 147

6

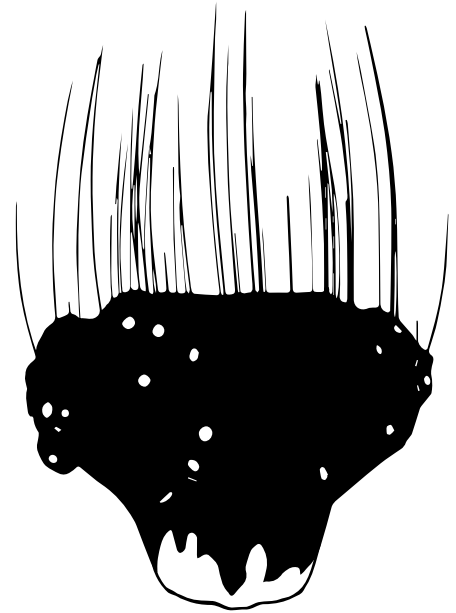
"Each partner had to be able to do something that the other couldn't achieve on its own."

Merlin Sheldrake, *Entangled Life* [2020], S. 97

7

"When he held a flame up to a mushroom, different mushrooms within the cluster responded with a sharp electrical spike."

Merlin Sheldrake, *Entangled Life* [2020], S. 71



How have these facts influenced our game and how have they been incorporated?

1

Sharing resources with the plants connected to your mycelium is a crucial part of our game play. The mycelium connects with plants in its surroundings and remains in constant exchange of resources with them.

2

5

This information helped us to decide which resources we would like to implement for our interactable plants.

3

The players can use specific upgrades to spread poison among the entire mycelium to poison the enemies which feast on them.

4

The connections with the most important plants, the resource providers, are connected by slowly growing, thickening roots to symbolize the strength and the importance of said connections.

6

The players should have to split tasks and be dependent on each other. Managing a mycelium can not be done by a single person.

7

As soon as the enemies attack the mycelium shivers. This effect represents the electrical spike that is caused by this alarming event.

+

Of course we had many more ideas that were inspired by this book. Unfortunately those couldn't be implemented due to the deadline. For example tripwires to trap the enemies, parasitic fungi upgrades, weather that influences the growth of mushrooms, plants that provide protection from the weather conditions and many more.

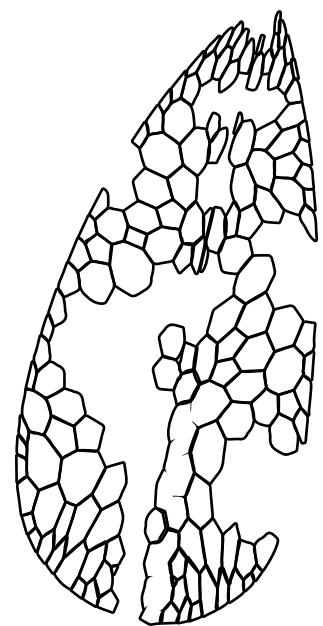
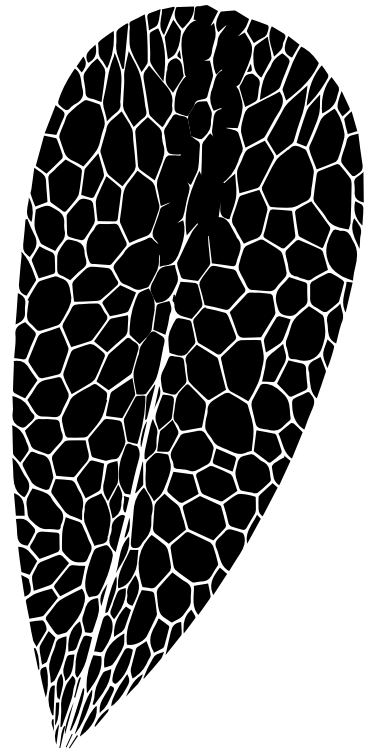


## 3.4 Exobiology

*We were inspired to conduct an in-depth research on a few keywords we came across while reading Merlin Sheldrake's book. One of those terms was "Exobiology". In "Entangled Life", he described a phenomenon related to this field of research.*

As humanity began sending scientists and robots to space, there was a widespread fear of space organisms reaching our atmosphere and damaging our ecosystem. Joshua Lederberg was one of the first scientists to highlight the dangers of dormant spores or bacteria that could potentially exist in space, be present in soil samples, or hitch a ride to Earth on people's clothing or equipment. These spores or germs could become activated upon encountering less hostile conditions, such as moderate temperatures or protected areas shielded from ultraviolet radiation, similar to those found on our planet. This concern prompted the implementation of strict security protocols for individuals or objects returning from space. Some scientists even suggest that fungi may have traveled through space on microscopic dust particles, and upon reaching Earth's surface, where different conditions allowed them to thrive, they "awakened." Over time, this process could have laid the foundations for our ecosystem. Even though our team agreed not to explore the space-fungi concept, primarily due to its extensive coverage in science fiction movies and our desire to maintain the non-human aspects of the game by avoiding a traditional storyline, we were still intrigued by the metaphorical elements of this theory. In a way, Multifarious places its players in the role of an alien, requiring them to adapt and survive in an unfamiliar environment. They must navigate unfamiliar conditions and learn to understand a new body, similar to how fungi spores wake up after traveling through space for thousands of years.

Pubmed: Germs in space. Joshua Lederberg, exobiology, and the public imagination, 1958-1964, <https://pubmed.ncbi.nlm.nih.gov/12198792/>, June 2002  
Nasa: Viking 1 & 2, <https://mars.nasa.gov/mars-exploration/missions/viking-1-2/>, 12 April 2018  
Quick Biochemistry Basics, Lederberg and Tatum experiment, <https://www.youtube.com/watch?v=-muO44bcexBo>, 2018



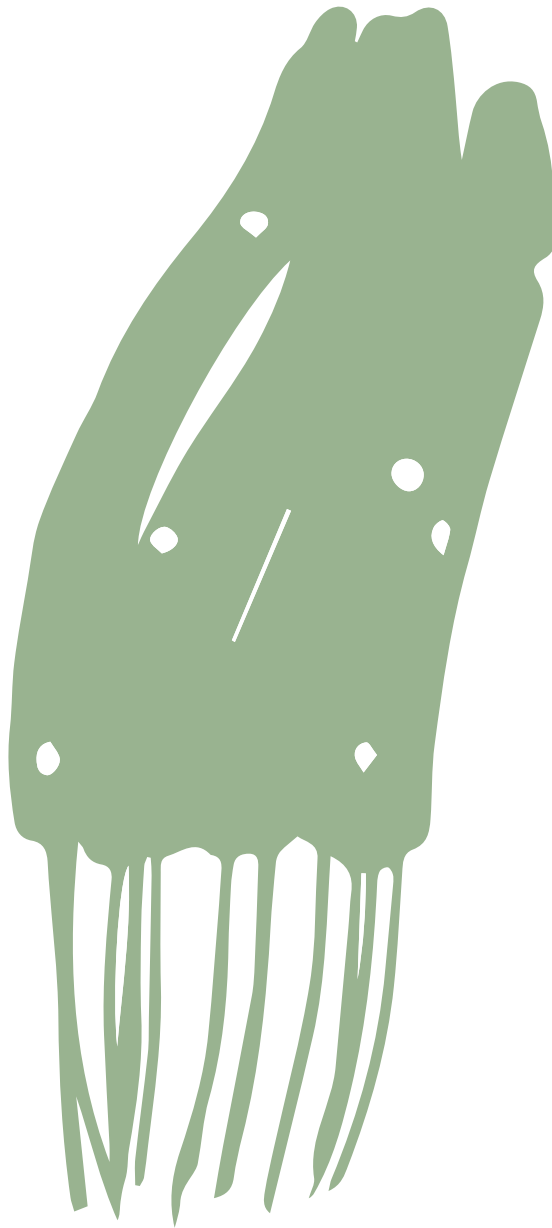
## 3.5 Refined Questions

*After our intensive research phase, we adapted our question to our newly accumulated knowledge and created a new set of new questions to define the further course of action.*

1. How do fungi act and how could this behavior be implemented in the game?
2. How could the growth of the mycelium network be designed to make it feel like natural growth instead of building?
3. What mechanics help to make players feel like a part of the same living network instead of the ones controlling it?



## 4.0 PROCESS



# 4.1 Process

*In this section, we will showcase the various steps and iterations we went through that ultimately led to the final version of Multifarious.*

## ◦ Overlap between Tech and Art

When determining the visual representation of fungi and a mycelium network in the game, we encountered numerous challenges that blurred the lines between art and technology.

Our goal was to create a mycelium network that could grow and expand in a way that felt natural to the player. However, the procedural generation of mycelial roots and vines as a 3D mesh is inperformant. Real-time generation of such a complex mesh was not feasible within the constraints of the game.

Additionally, pre-generating the mesh and activating specific parts as needed was not an option since the growth of the mycelium had to be entirely dynamic and influenced by the player's behavior. Therefore, we needed a solution that allowed for realtime generation of the mycelium network during gameplay.

After careful consideration, we determined that the most practical approach was to "draw" connections and vines onto a render texture. This rendered texture would then be evaluated in a shader to modify the visual appearance and surface texture of the terrain based on where the mycelium had grown and where it had not.

This initial concept underwent further development from both the technical and art perspectives and served as the foundation for the final implementation of the mycelium network in the game.



## 4.2 Process - Art

*From the beginning on we had quite a lot of ideas of what our visual style should do and achieve. These were the guiding principles that shaped our visual process and were further refined through iterations involving various art styles, techniques, and ideas.*

- *Starting points and requirements*

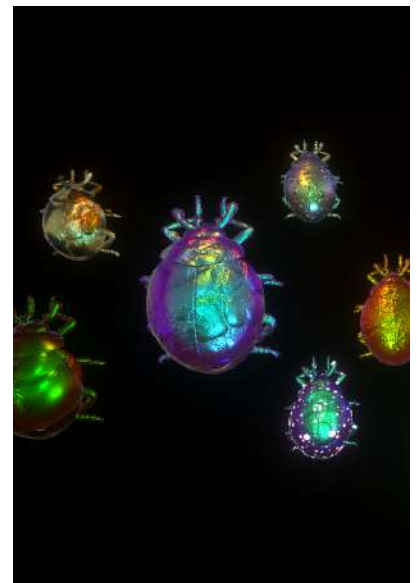
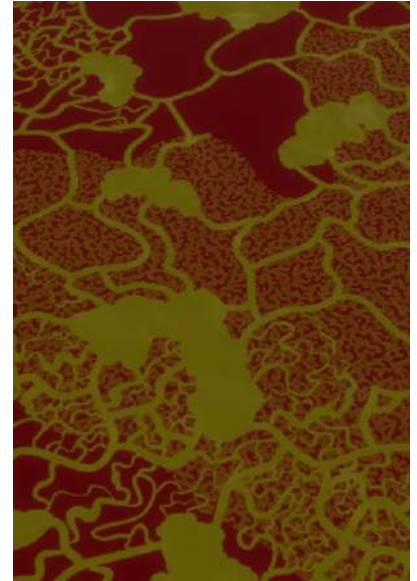
1. Our game should be accessible to people who suffer from motion sickness or migraines. Since we both struggle to play games that are not optimized for issues like such, we wanted to make sure that our game is playable for everyone and does not cause us to feel sick while we are working on it. We wanted to achieve this effect by paying close attention to nervous visuals and important player components like the camera settings.
2. The visual design should support our mechanics and enable players to comprehend and understand them effectively.
3. Performance was a crucial aspect of our design process, and it was taken into account from the outset. Regardless of the ideas we explored, we were aware that the game needed to run smoothly so that the visuals wouldn't get in the way of game experience.
4. The spirit of the mysterious and abstract world of fungi should be transferred into the final look.



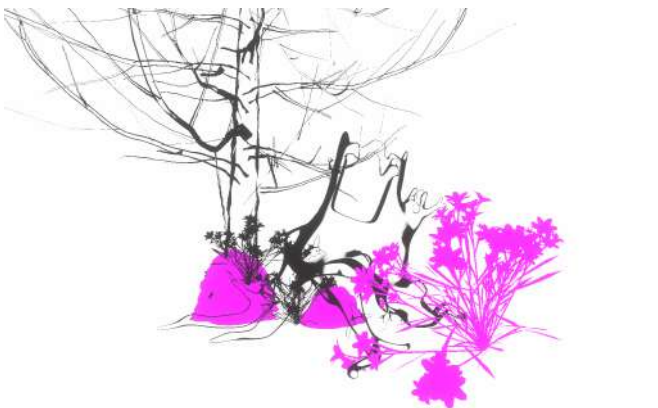
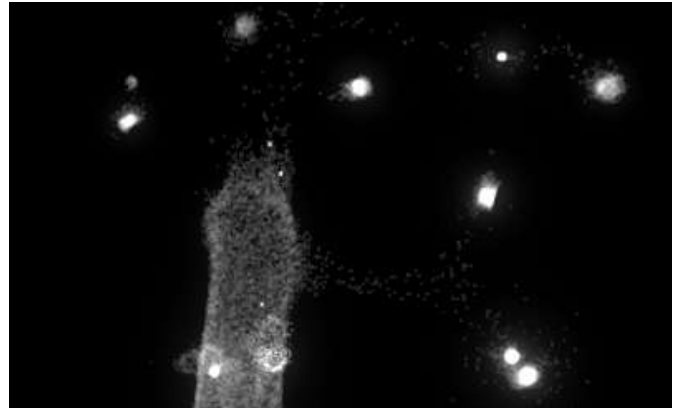
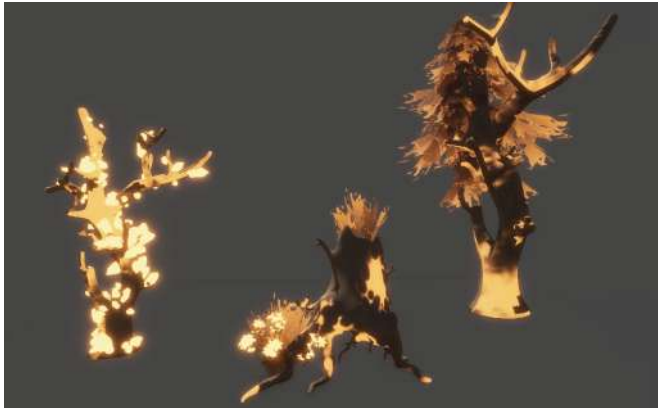
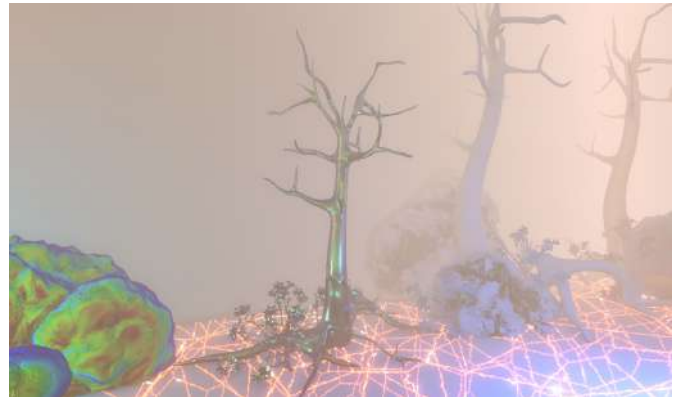
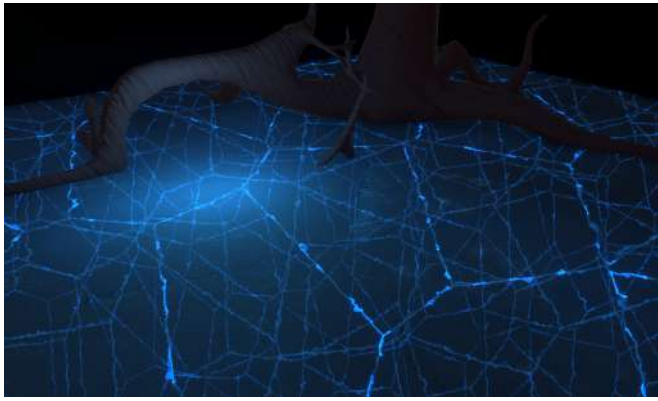
- *Iteration 1 – Holistic research and experiments of what can be done*

At the outset, establishing an overview of the visual and technical possibilities was important. Especially knowing that designing such a project will be a great challenge and will ask for alternative solutions, having a repertoire of knowledge about the state of the art comes in handy. Even if we felt like something was irrelevant or might not be useful for this project we looked into it. Given our uncertainty about the direction we would ultimately take, we wanted to ensure that we wouldn't overlook any opportunities by lacking knowledge about a suitable method later on.

While exploring the various shader techniques in Unity, we came across numerous intriguing effects and attempted to apply them to our overarching theme, the fungal universe. For instance, inspired by a frozen lake shader by "Binary Impact", we utilized the Parallax-Effect to create an underground roots system. Visualized through small glowing dots, the viewer gains the impression of nutrients being transported through the roots. Additionally, we delved into metallic materials, holographic simulations, and point clouds. By adjusting the values of those shaders, we achieved several fascinating results. By accident, we even created a thermal imaging camera effect. Although we liked some of these art styles, they all encountered similar issues. They appeared experimental and captivating, but were difficult to read and strained the players eyes, which was something we aimed to avoid. While they might be suitable for poster designs or static visuals, they seemed impractical for a dynamic project like a game.



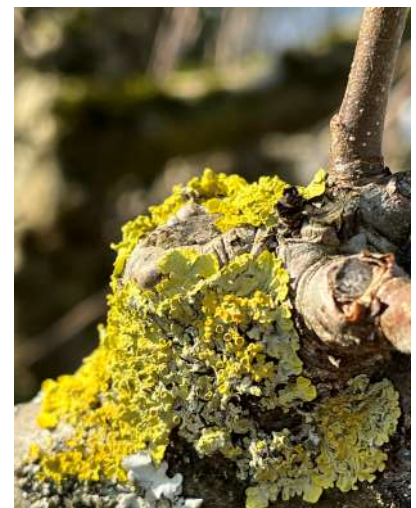
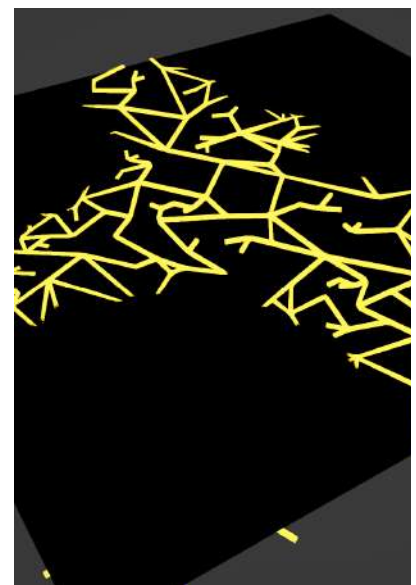
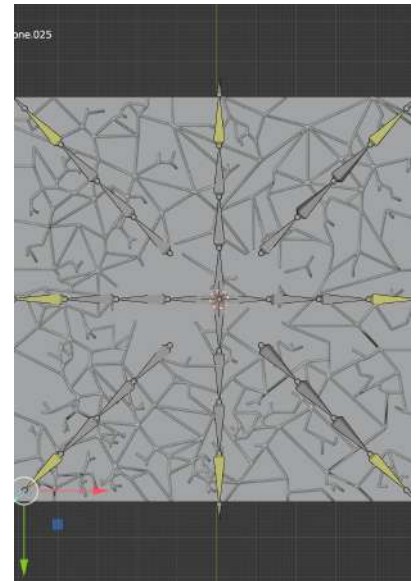




## ◦ *Iteration 2 - Searching for design solutions*

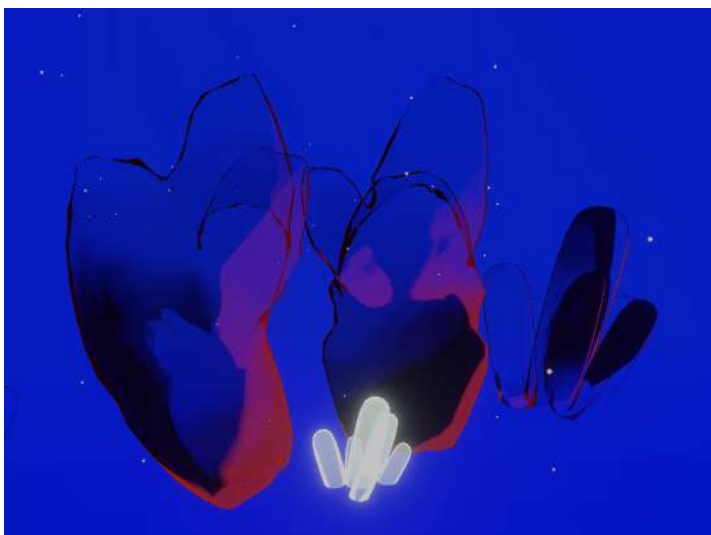
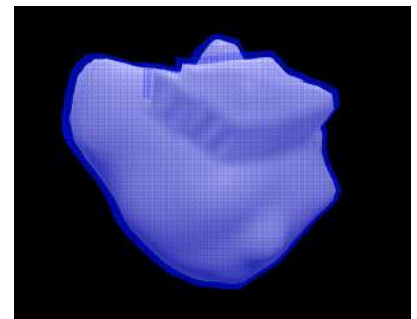
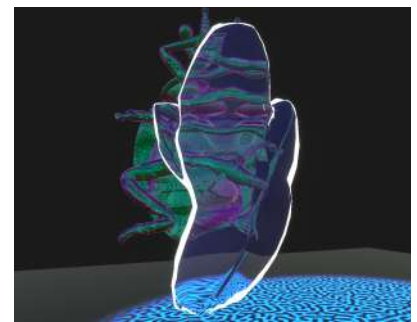
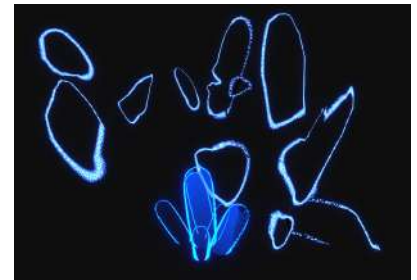
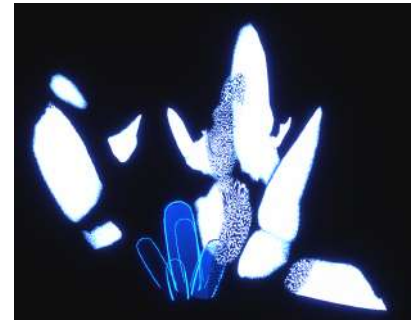
A new challenge that emerged during the first phase of our process was the design of the roots. In iteration 1, we initially planned for the mycelium to be a 2D effect, but the outcome was disappointing. It felt lifeless and more like a piece of paper rather than a living organism engaged in constant exchange with its surroundings. Even the Parallax Effect failed to fix this issue and displacement was not up for debate at this point of the project. Mainly due to the amount of vertices needed for a smooth displacement result. However, out of curiosity, we tried it to see if we liked the result, and it turned out to be the effect we wanted to achieve. One idea we had was to sculpt a 3D tileable mesh for the roots and rig it so that we could partially drag it through the floor based on the player's position. This approach yielded a result close to what we were trying to achieve, but it introduced several issues. One problem was the flickering that occurred in areas where the floor and roots intersected. Additionally, creating an organic-looking 3D model, which required a reasonable number of vertices, proper UV unwrapping, and weight painting, that could not be generated by an algorithm add-on due to the need for tileability, was a tiring obstacle to overcome. While it would have been possible for sure, when thinking of the unpleasant side effects this method would create and how unsatisfied we would be with the end result, we decide to take a different route.

We were hung up on trying to find a good solution for the mycelium's visualization. As a result, we decided to shift our focus to the environment and establish the overall look first before tackling the details. We took walks in the forest, taking pictures of mushrooms, locations, plants, and structures. The internet served as a great source of inspiration as well. We explored various styles influenced by the collected material. While the newly developed art styles appeared intriguing, they didn't fit well. The transparent materials used caused display issues when overlapping, and since they were developed without considering the mycelium, they seemed to have somewhat of a main character feel to them, making the mycelium feel insignificant and negligible in comparison.





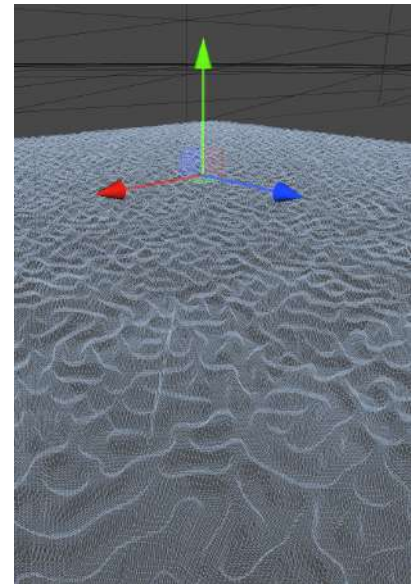
The mycelium is the most important part of our concept so it should stand out the most. The idea of exploring the senses of sight and touch as key elements of our visual style intrigued us, considering that fungi likely perceive the world differently from humans. We figured that touch may be the closest out of all human senses to a fungi's perception of its environment. While we were captivated by this concept, our attempts to visualize it proved challenging. Many of those designs were hard on the viewers eyes and hindered player's navigation due to limited visual information presented. This would have compromised our game mechanics, which was not acceptable. However, our experimentation with transparent materials led us to explore Unity's fog feature. By adjusting the fog distance, we discovered that it had an interesting effect on the partly translucent objects, allowing the environment to be seen through them. This enabled objects to appear unexpectedly while still maintaining a certain amount of visibility through masked, opaque areas on their surface. We liked the idea of having different colors inside the base compared to the outside world, which are influenced by the fog's values changing as players venture beyond their base. Ultimately, we decided to keep this concept in mind for the environmental design.



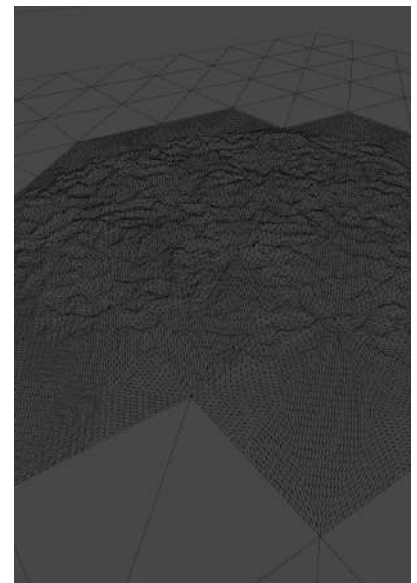
- *Iteration 3 – Back to the roots*

After experimenting with the environmental objects and gaining a clearer vision for their design, we shifted our focus back to the network's visuals. We recognized the importance of defining the design for the roots and players first before creating the design of the environment. In our previous attempt, we had done it the other way around, resulting in numerous styles that did not suit the structure of the mycelium. However, we had learned from that experience.

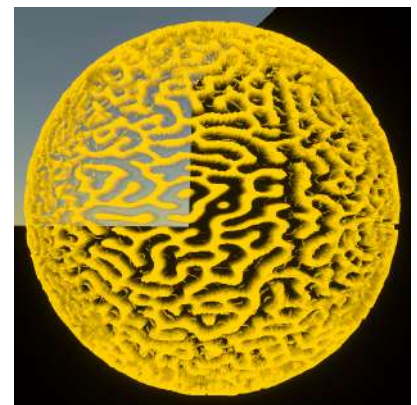
We couldn't let go of the idea of having a three-dimensional fungi network, so we embarked on another attempt. This time, successfully. We came across custom tessellation shaders. Using this approach, we could use a floor plane with a low amount of subdivisions and dynamically add more detail in the areas where displacement was needed. By displacing the roots from a texture and leaving away unnecessary details, we were able to achieve the desired effect of our wobbly 3D roots.



Displacement



Masked tessellation



## ◦ *Iteration 4 – Finalizing the art style*

As we were sure we would be able to create the effect we desired, it was time for us to decide what exactly we wanted and how the game's visual design can achieve that. Redefining our design pillars was the first step we took to define the final art style.

### Readability and visual communication

---

Fungi are neither plants nor animals. They are complex, alien-like beings. To us humans, a fungi's interpretation of its surroundings might seem abstract and hard to grasp. Therefore the visual communication should inherit a certain amount of relatability and familiarity, to make sure that the players are challenged but not overwhelmed. Yet still, the fungi's characteristics and otherness are important elements of our visual and mechanical design.

### Immersion and player motivation

---

Multifarious allows its players to dive into a mysterious and unfamiliar world which would normally not be accessible to us. The Game-Play should feel satisfying and wants to motivate the players to engage with their Fungi-Network and its surroundings. The spositive reinforcement when interacting with the environment should be established by the visual effects, the movement of the mycelium and its animations. The element of speed and the possibility to improve those satisfying skills will support said elements.

### Player teamwork

---

Teamwork is an important keyword when talking about Fungi. Due to earth's initial merciless survival-conditions and their incapability of harvesting certain life-sustaining components, Fungal-Networks became fantastic Team-Players. Multifarious wants to organically motivate its players to work together, split tasks and enter a symbiosis with the game's environment by allowing both players to make decisions for each other, share the resources or upgrades, and limit the players accessibility to certain actions.



We implemented the idea that the outside world should look completely different in comparison to the inside of the mycelium. Originally the concept was planned to be designed in red and blue because according to Merlin Sheldrake, those are the colors some fungi can actually sense. But not only is this a very aggressive color scheme, we also figured that applying a color code that needs no specific knowledge to be understood would improve our concept. From a fungus's perspective the underground is where it is at home, where the world is a safer place. Being above ground increases the chance of being eaten or to be exposed to other potentially dangerous factors like harsh weather conditions. Therefore the inside of the mycelium is designed in a darker nightish color scheme while the outside world is visualized in a much brighter tone. We believe that this color arrangement indicates that the player-character is not human and supports the alien-like aspects of the game for the reason that our human intuition has different feelings towards those previously mentioned light situations. For humans it would likely be the complete opposite. The forest is a good example for that. At night it feels ominous and dangerous but during the day it feels like a great place for a picnic. Furthermore the brightness symbolizes the undiscovered. The blank paper if you will. Which is ideal for an environment that wants to be explored. Therefore everything that becomes a part of the mycelium network throughout the game turns dark while the outside world and its belongings remain bright.

Our design concept determined what our shader has to be able to do in order to meet these requirements. But still it was constantly updated as we kept coming up with additional ideas for the concept while we were working on the final product.

Our shader had to ...

... have customizable tessellation that is maskable by distance and the base location to avoid performance issues.

... receive all sorts of light sources and inherit their colors in the affected areas to enforce a wider color range and diversify the brightness values which help to dissolve the monotony of repetitive textures and create a more immersive atmosphere for the game's environment.





... receive specular lights to add a satisfying gloss on top of the roots which define the mycelium's texture and visually elevates the roots from their background. The mycelium's grainless, wobbly and seemingly slippery surface is necessary to emphasize the fact that the roots are a part of a living organism.

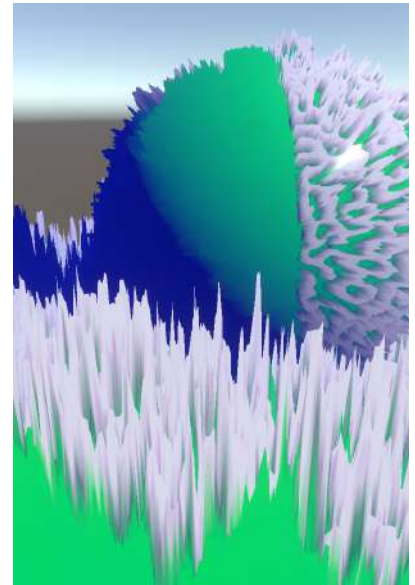
... do displacement that is masked by numerous render textures and several different seamless root textures which are animated by a scrolling noise.

... work with normal maps to avoid the necessity of recalculating the normals so the light can be displayed properly.

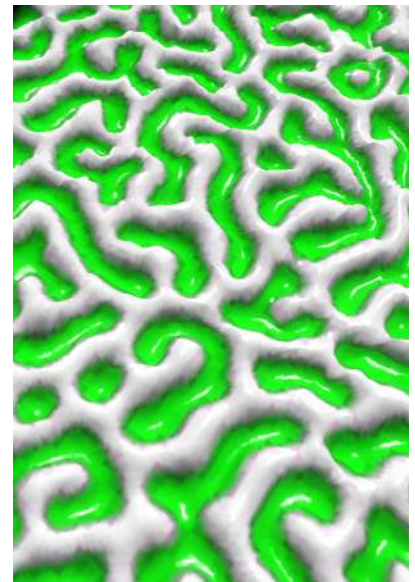
The shader had to be scripted and could not be done in the Unity Shader Graph because it is not possible to interfere with the tessellation state inside the Shader Graph tool when using the Universal Render Pipeline. Switching to the High Definition Render Pipeline, where tessellation can be enabled for Unity's shader tool, would have created new and potentially bigger problems. For example, issues that might occur due to the HDRP's young age. Such problems can be hard to resolve since they are hardly documented. Therefore, staying with the URP was the fitting option for this project.

A lot of elements, such as colors or effect strength values, needed to be separated and independently adjustable from each other so that they could be highlighted and displayed correctly. This was necessary due to the high color and brightness contrast between the outside world and the elements inside the base. While certain effects and colors may look good in one place, they may not work well in both locations. Moreover, specific elements, like the players, had to be customizable separately from the rest of the mycelium to highlight them as a distinct part of the network.

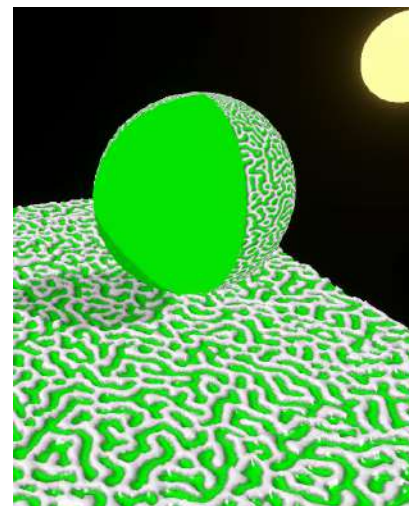
Since there are numerous non-static components that influence the appearance of the mycelium, all of these values needed to be properly masked in case they overlapped. For instance, the growing connection of a plant should not be added on top of the main mycelium, even if its connection starts in the center of the base. This would not only cause the values to change but also result in an



Specular light without normal map



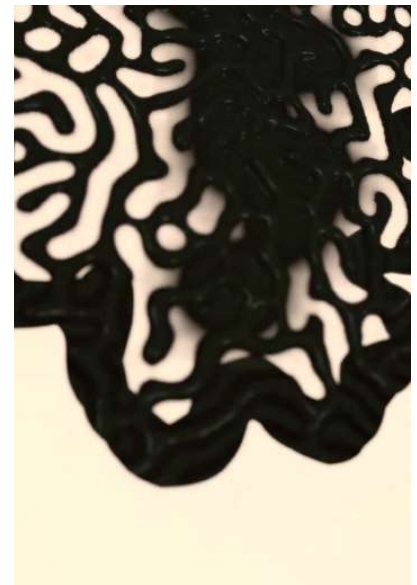
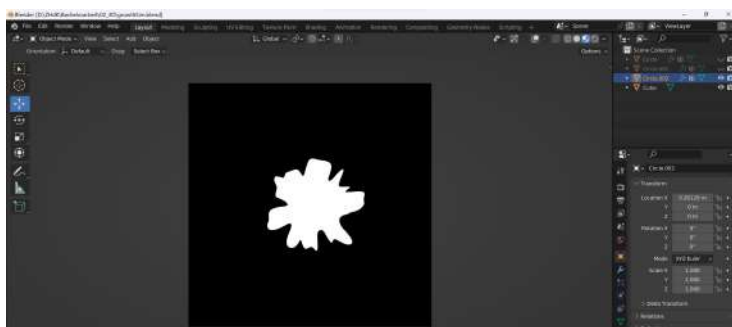
Specular light with normal map



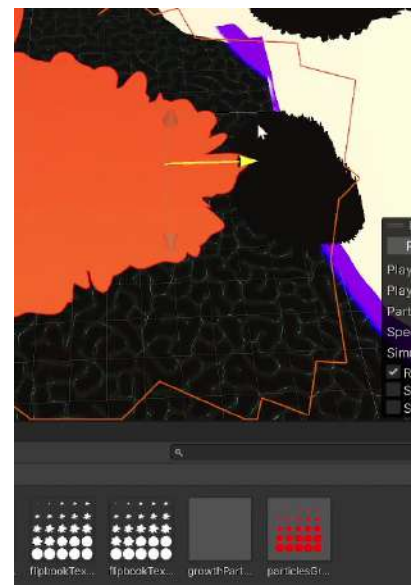
unpleasant cut when the two textures are baked together as the connection is done growing.

As mentioned before, our shaders' tessellation can be masked by distance, textures, and vertex colors. However, as the visual concept evolved, we found that the vertex and texture masks became obsolete. We decided that the players should be accompanied by their „exploration goo“ wherever they go. Consequently, the displacement now needs to occur wherever the players are positions are, rather than being limited to the inside of their base.

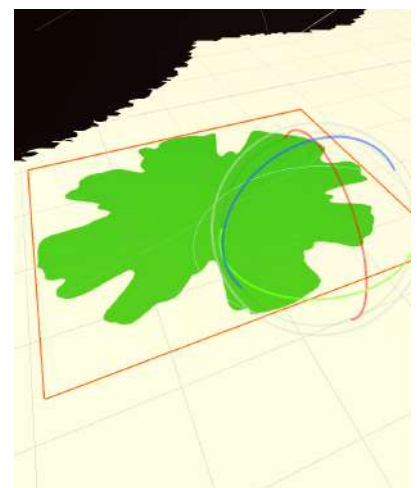
We utilized various types of particle systems throughout the project. Some of them were passed on to the shader for displacement and visual masking. The smooth and satisfying movement aspect was of great importance to us, which is why we dedicated a lot of attention to the animations used for the particle systems. For these animations, we initially created hand-drawn flip book textures to simulate the spread of fungus mycelium across the floor. While this served well for testing the effects, we noticed there was room for improvement. To achieve better growing animations, we downloaded a Blender add-on called „blender-differential-growth“ by „Inca“. This algorithm effectively simulates organic growth for plants. To utilize the add-on, we applied a white material to the desired plane and recorded the simulation from a top-down perspective. We then selected the best frames, made some adjustments, hand painted over them, and converted them into flip book textures using Photoshop. Although we also experimented with Houdini, the Blender method proved to be more efficient and produced better results. Through this technique, we achieved the final growing effect that was implemented in the game.



Exploration goo



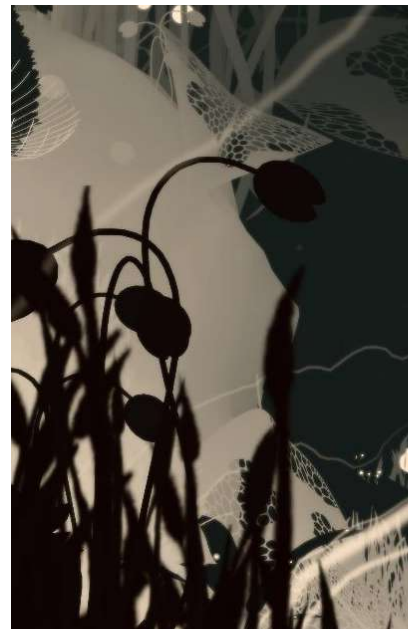
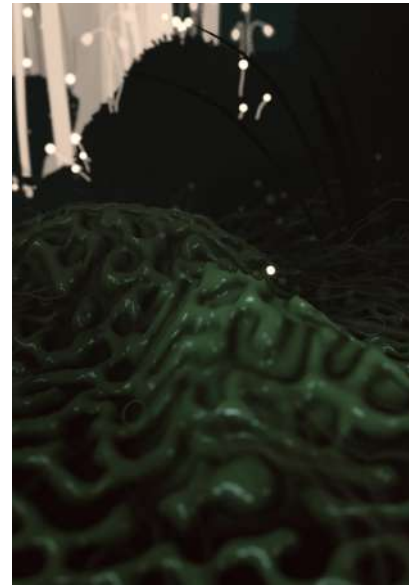
Exploration goo particle system and flip book textures



## The environment

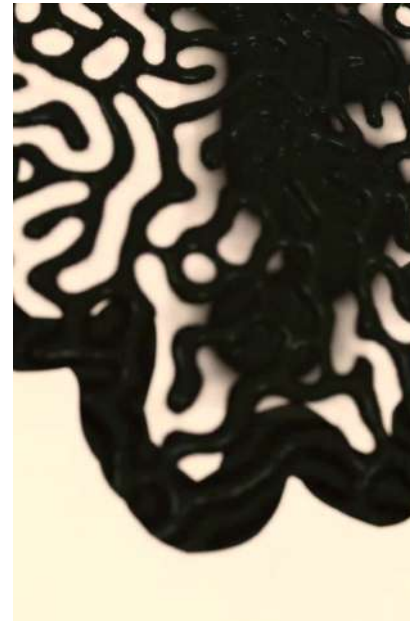
The environmental concept should make the players feel small. They should find themselves in an unfamiliar world. All of the environmental elements need to be able to change their appearance from black to white as soon as they become a part of the mycelium. While it's important to add enough detail to create visual noise in the outside world, plants that don't provide any resources or updates should feel distant or anonymous, like a person wearing a mask. This conveys that they are neither interactable nor capable of forming a physical or emotional connection with the mycelium. This not only supports readability, which is important to us, but also enhances the feeling of being an alien. It's similar to attending a party out of politeness and feeling stuck with people you have nothing in common with. Everyone already knows each other, and being there is super awkward until you meet someone at 3am in the kitchen who currently has the same weird hyper-focus like you do, and you talk about it until the sun rises.

We tried achieve this effect using an unlit shader material which only displays their silhouettes as they are not capable of receiving any lights and with a little help from the fog feature the details of their shape become visible. With this method they become easier to read and can be told apart from their background.

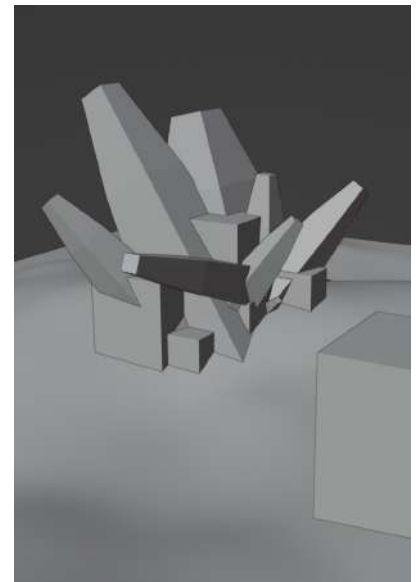




During playtesting, we noticed that certain types of plants had a particularly satisfying effect when changing colors. These were mainly small or thin plants that were placed in big bulks. As the mycelium formed connections and their colors switched to black, the trails left behind by the growing mycelium became visible from a distance, allowing players to see the changes they had caused in a more noticeable way. To facilitate this, we intentionally included a grass forest on the map and scattered various types of tiny and thin plants across the terrain. Additionally, small plants sprout from the new mycelium connections to create a similar effect and showcase the fertility of the newly formed partnerships.

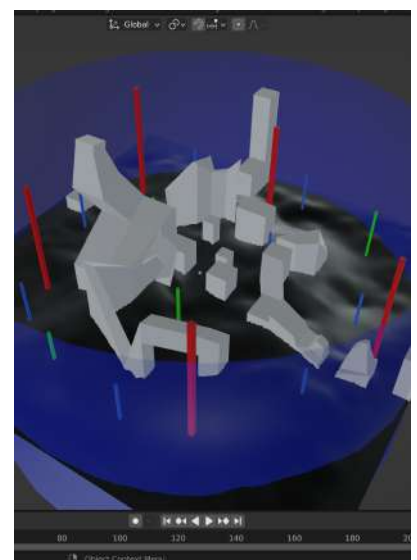


Another recurring feedback we received was that people enjoyed the wobbly movement and visuals while exploring. To ensure this aspect wouldn't be lost in the process of creating the environment, we deliberately created areas with lower density of objects, almost resembling a race track in a way.



Gray boxing in Blender

To delineate the different areas, we created a "weight map" with color coded placeholders. This allowed us to define which areas needed their own distinct "theme" and memorable objects for players to latch onto when trying to remember their location.



Weight map



## 4.3 Process - Coding

- *Dynamically expandable base*

*The biggest challenge was the mycelium network, acting as the player's base, which had to be dynamically expandable while also fulfilling numerous mechanical roles.*

### Expanding the base

To achieve the aforementioned solution of applying the mycelium network to a render texture, we started with prototyping focused on this underlying system. Initially, we placed the network on the green channel of the render texture and used it to visualize the system by applying the texture to the ground.

\*1 In this version, the player could expand the base by simply holding a button on the gamepad and "drawing" on the floor. We achieved this by calculating the player's position on the texture using their world position.

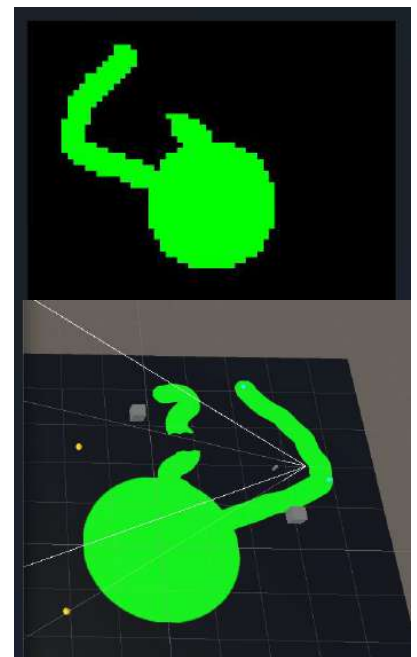
### Connecting and disconnecting of Resources

\*2 The game heavily focuses on connecting and forcefully disconnecting resource nodes or potential upgrade nodes. However, this presented a new challenge in determining if something was connected to the center of the network. We relied on reading the render texture to determine if the colored pixels formed an uninterrupted connection from the center to the desired node.

To address this challenge, we implemented the Dijkstra pathfinding algorithm, which was applied to the texture. Each pixel on the texture represented a tile on the pathfinding grid. By using this method, we were able to regularly check if each individual node on the map had formed or lost its connection during the pathfinding calculations.



\*1



\*2

## Automatic connections instead of drawing connections

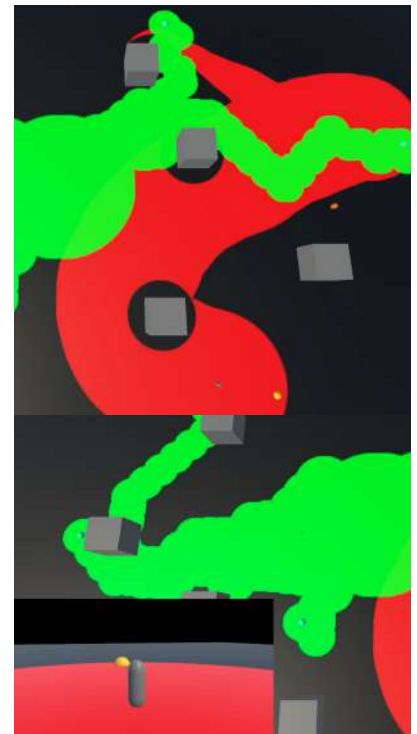
The network was now expandable through “drawing”, but it lacked the organic growth and autonomy we aimed to achieve. Our inspiration came from observing how yellow slime mold spreads in search of food and forms connections with newly discovered sources.

To capture this essence, we experimented with different variations of what we referred to as “Exploration Goo”. The initial version involved activating and spreading the Goo by holding a button, releasing it to propel the player in their desired direction towards the border of the expanded Exploration Goo. However, we quickly abandoned this approach as it proved to be tedious and tiresome, requiring repeated button presses.

Instead, we implemented the Goo as a continuous companion that moves with the player, leaving behind a slowly decaying trail. If the Exploration Goo established an uninterrupted connection from the base to a connectable node, the base would utilize the available information, including the areas it had already connected and the regions discovered through the Exploration Goo. The rest of the world remained unknown to it. Expanding on our existing Dijkstra pathfinding algorithm, we incorporated the newly added Exploration Goo into the calculations and stored data. This allowed the mycelium network to identify the most optimal path to discovered nodes on the provided pixel grid.

As a visual representation, we introduced dots on the render texture indicating the optimal path to the next node, enabling the expansion of the mycelium network and connection to various nodes.

The Exploration Goo is depicted in red on the floor, while the mycelium network is represented in green. \*3



\*3

To create smoother and gradually forming connections, we introduced a particle system that follows the path identified by the Dijkstra pathfinding algorithm, leaving behind a growing trail. This particle system's output is captured by a camera positioned above the scene, which sends the image to a render texture. This render texture is then utilized for calculations and shaders while the connection is being formed.

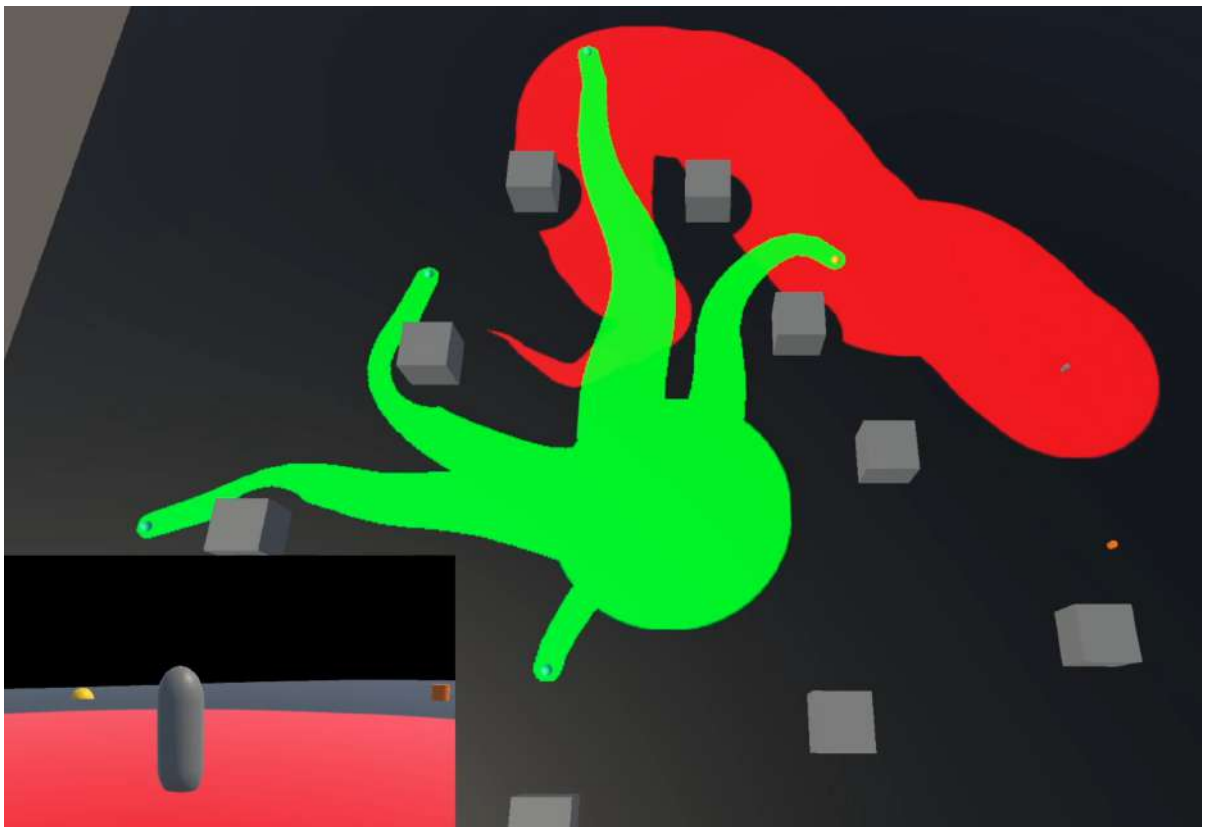
Once the connection is complete, the render texture is extracted and the green channel is

stamped onto the floor's render texture, establishing a permanent and interactive connection. Additionally, we incorporated square boxes with black circles around them as obstacles. These obstacles are represented as sprites overlaid on the map and are taken into consideration during the pathfinding calculations. The camera filming the new green connection particle system and the red Exploration Goo particle system also captures these obstacles for later use in the pathfinding process. \*4

Mycelium networks exhibit high efficiency in their growth and connection formation to food sources. In the screenshot provided, we can observe some unnecessary movements in the forming connections. To address this, we improved the path calculated by the Dijkstra pathfinding algorithm by converting it into a smoother path using bezier curves. This enhancement was achieved by implementing the DeCasteljau algorithm. By utilizing bezier curves, the mycelium network in the game was able to form seamless connections to newly discovered nodes, following the paths revealed by the player's Exploration Goo. This final version of the network was implemented and used in the game. \*5



\*4



\*5

## ◦ Performance

*With all these pathfinding calculations happening continuously at runtime, since the players will always be moving around discovering new nodes to connect to, the game had to handle a large amount of calculations that couldn't fit into single frames.*

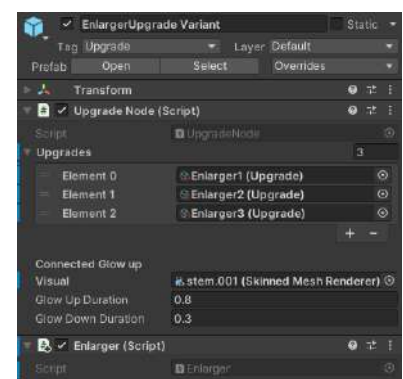
The first thing we did was utilize async operations for most of these calculations wherever applicable. By using the UniTask async integration, we managed to put all the pathfinding calculations on their own threads, triggering events when the latest data became available for all the objects in the scene that relied on it. However, certain graphic interactions with the render textures themselves couldn't be offloaded to separate threads.

A second step to optimize the calculations was downsampling the render texture. Since the texture didn't need to be as high resolution for pathfinding as it did for the shader, we reduced its resolution by half four times before sending it for evaluation. This implementation also made use of async operations, although it couldn't be placed on a separate thread as mentioned earlier.

## ◦ Expandable Upgrade System

*The goal of the upgrade system was to ensure its expandability and flexibility, considering that the full scope and variety of upgrades was not yet clear. Additionally, a key objective was to provide a wide range of diverse upgrades to empower the player with meaningful choices during the selection process.*

Each upgrade node in the game offers three choices for the players to select from. These choices are implemented using scriptable objects of the "Upgrade" type, which are added to the respective node. Additionally, each upgrade has a reference to its visual representation, allowing for visual effects when the node is connected to the network.

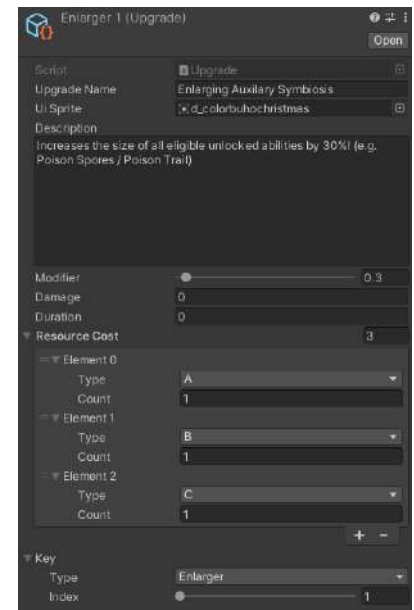


Furthermore, each upgrade node has an individual script attached to it, responsible for handling the specific functionality of the upgrade itself. This approach enables the customization and unique behavior of each upgrade node in the game.

Each upgrade scriptable object encompasses comprehensive information for its display in the user interface, such as the name, description, and sprite. Additionally, it contains data related to the functionality of the upgrade.

This functionality data includes the modifier associated with the upgrade, and if applicable, it may also include values such as damage numbers and duration. The resource cost required to activate the upgrade is also specified within the scriptable object.

Furthermore, the scriptable object holds a key that consists of the upgrade type and its corresponding index. This key is crucial for the individual upgrade script to determine which of the three options it needs to activate or deactivate. This allows for proper selection and management of the upgrades within the game.



```

public interface IUpgrade
{
    6 references
    public void EnableUpgrade(int index, PlayerStateManager playerIndex);
    5 references
    public void UpdateUpgrade(int index, PlayerStateManager playerIndex);
    8 references
    public void DisableUpgrade(int index, PlayerStateManager playerIndex);
}

```

Each individual upgrade script in the game implements the IUpgrade interface, allowing it to be activated through the upgrade node. The parameters required for activation include the index of the selected upgrade and the player intended to receive the upgrade.

By utilizing this interface, upgrades can be enabled, disabled, or updated based on changes in the game state.

Passive upgrades primarily involve adding or removing values from modifiers in various affected scripts. For instance, a passive upgrade may increase a player's movement speed by adding 0.3 to their movespeed modifier, resulting in a 30% speed boost. Some passive upgrades may have more complex implementations, but their core functionality revolves around modifying values or enabling specific interactions within a script.

On the other hand, activatable abilities are more intricate. These abilities can be activated from the base center when entering the defense mode. To simplify the activation and interaction process between scripts, an additional interface called `IDefenseAbility` was created specifically for active abilities.

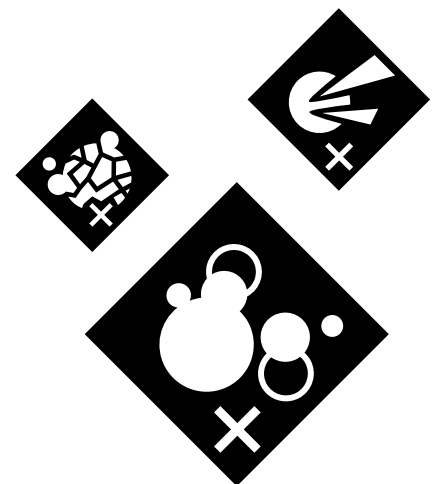
```
11 references
public enum DefenseType
{
    Projectile,
    Immediate
}

10 references
public interface IDefenseAbility
{
    4 references
    public DefenseType type { get; }
    5 references
    public Sprite sprite { get; set; }
    5 references
    public void ActivateAbility(PlayerStateManager player);
}
```

The `IDefenseAbility` interface provides a choice between two types of active abilities: those with direct effects upon activation and those that manifest as controllable projectiles. It includes a sprite that represents the ability in the UI, as well as the "ActivateAbility" function that contains the core code responsible for defining each ability's behavior.

For each ability that can be learned by activating the corresponding upgrade, a script implementing the `IDefenseAbility` interface is created. This script is attached to the player when the ability is learned and removed when the upgrade is deselected.

During base defense, all scripts attached to the player that implement the `IDefenseAbility` interface are gathered into a list and prepared for selection in the UI. This approach ensures flexibility and expandability, as the entire interaction between abilities and the game system is facilitated through the implemented interface.



- *Enemies*

To facilitate the addition of new enemies or enemy behaviors, a state machine was implemented. This allows for easy management and control of enemy AI.

The enemies in the game were created using Unity's NavMesh system, which provides navigation functionality. Additionally, a custom solution was developed to ensure that the enemies remain aligned with the floor at all times. This is particularly important since the game features insects as characters, which require different types of motion compared to humanoid characters.



## 5.0 GDD





# 5.1 GDD

## ◦ Project Overview

*A mycelium is trying to survive and eventually thrive through expanding and entering a symbiosis with the forest around it.*

### Pitch

The players will each take control of a small mobile part of the same fungus network and try to connect it to various natural resources, expanding the mycelium in the act. Beware of the natural predators which will periodically find their way to the fungus and try to feed on it if not fended off in time.

### Specifications

Title: Multifarious  
Developers: Anja Irniger and Luca Imesch  
Music & Sound: Reina Burkhalter  
Genre: Roguelike, Survival  
Play Mode: Multiplayer  
Platform: Windows & Mac  
Engine: Unity



### Readability and visual communication

---

Fungi are neither plants nor animals. They are complex, alien-like beings. To us humans, a fungi's interpretation of its surroundings might seem abstract and hard to grasp. Therefore the visual communication should inherit a certain amount of relatability and familiarity, to make sure that the players are challenged but not overwhelmed. Yet still, the fungi's characteristics and otherness are important elements of our visual and mechanical design.

### Immersion and player motivation

---

Multifarious allows its players to dive into a mysterious and unfamiliar world which would normally not be accessible to us. The Game-Play should feel satisfying and wants to motivate the players to engage with their Fungi-Network and its surroundings. The spositive reinforcement when interacting with the environment should be established by the visual effects, the movement of the mycelium and its animations. The element of speed and the possibility to improve those satisfying skills will support said elements.

### Player teamwork

---

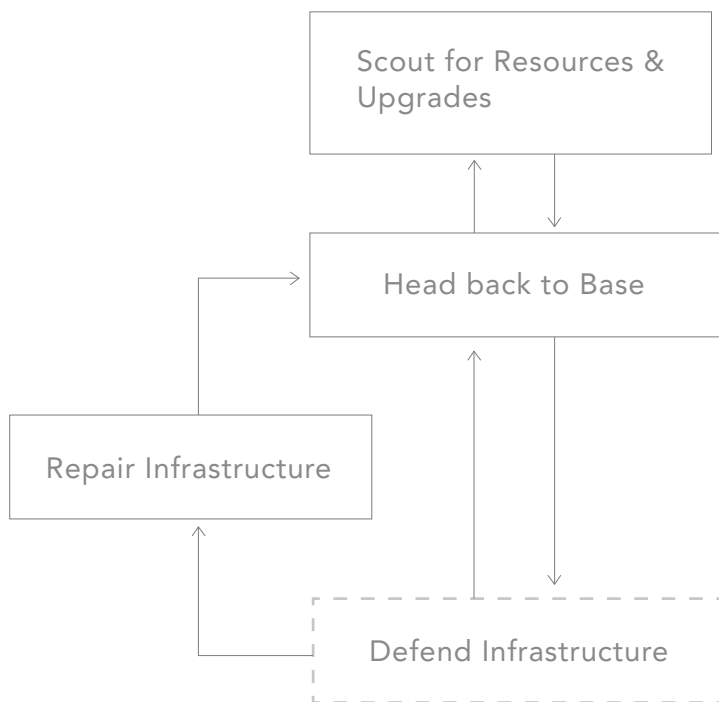
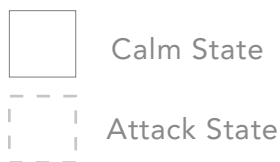
Teamwork is an important keyword when talking about Fungi. Due to earth's initial merciless survival-conditions and their incapability of harvesting certain life-sustaining components, Fungal-Networks became fantastic Team-Players. Multifarious wants to organically motivate its players to work together, split tasks and enter a symbiosis with the game's environment by allowing both players to make decisions for each other, share the resources or upgrades, and limit the players accessibility to certain actions.



- o *Mechanics*

### Macro & Micro Loop

The game unfolds within two main states: The "calm state" and the "get attacked state." During the calm state, players have ample time to explore, manage their resources, and upgrade their abilities. However, when enemies appear and threaten the mycelium network, the game transitions into the get attacked state. In this state, players must utilize their unlocked defense capabilities to fend off the enemies and protect their base. Once the enemies have been repelled, players can return to repairing any damage, continue exploring their surroundings, gather new resources, unlock upgrades, and activate them within their base.



- *Features*

### Mycelium Network

The central component that ties many of the game's features together is the mycelium network. Functioning as both the base and the players' body, the network is a key element of gameplay. Players have the ability to expand the network by discovering resource or upgrade nodes in close proximity to it. The network then establishes the most efficient connection to these nodes, utilizing the geographical information that the players have uncovered during their exploration. This dynamic expansion of the mycelium network allows players to strategically extend their reach and optimize their connections to various resources and upgrades.

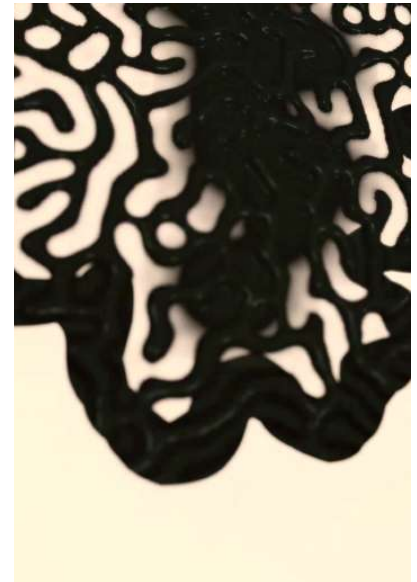
#### Exploration Senses

The connections within the mycelium network are established using the players' exploration senses, which are visually represented as a black net surrounding the player. As the player moves, this net expands, forming connections to discovered nodes. However, if the player moves too far from the base, the net gradually shrinks in size. To restore it to its full extent, the player must return to the base and reconnect with their mycelium. This mechanic is crucial because the mycelium network can only connect to resource and upgrade nodes that are directly linked to the network through the players' exploration senses. If the connection between the exploration senses and the mycelium network is lost, the network loses awareness of the player's movements and discoveries. Consequently, it becomes unable to establish connections to newly discovered nodes until the link is reestablished.

#### Shared resources

With both players being part of the same mycelium network they share all resources. They have to maintain their expanding mycelium by finding enough resources to accommodate the additional expenses of the bigger network.

If they can not sustain the network with their resources it will slowly start to decay. This process can be stopped and slowly reverted by finding more resources in time or by reallocating their resources from the chosen upgrades to the mycelium's needs.



Exploration goo

## Resources

There are three distinct types of resource nodes in the game, each providing a different essential resource for the thriving of the mycelium network. These resources are categorized as water, minerals, and energy.

In terms of importance and scarcity, water is the most vital resource for the mycelium network's survival, followed by minerals and energy. Water resources are the most abundant and readily available, while energy resources are the rarest to come across.



The significance of these resource types is reflected in the cost and requirements of acquiring upgrades. More advanced and powerful upgrades demand a higher amount of energy compared to weaker upgrades. Weaker upgrades typically necessitate a combination of water and minerals. This approach enables a balance to be struck between the number of strong upgrades that can be active simultaneously compared to weaker upgrades.

By carefully managing and prioritizing the acquisition of these resources, players can ensure the sustainability and growth of the mycelium network while making strategic choices about which upgrades to prioritize based on their resource requirements.

## Upgrades

Upgrades in the game are unlocked through a process of discovering and establishing a symbiotic relationship with unique plants or fungi found within the game world. When players encounter these special organisms, they are presented with three options for upgrades to choose from.

Once an upgrade is chosen, it can be activated at the center of the mycelium network by allocating the required resources that the new partner organism demands. Each player has the ability to activate their chosen upgrade individually, allowing for customization based on their playstyle or designated role within the team.



The resources needed to activate these upgrades are drawn from the shared resource pool that both players contribute to. This provides players with the flexibility to decide how to allocate their resources among themselves. They can choose to invest all resources into a specific player, distribute them evenly, or devise a strategy that best suits their gameplay objectives and team dynamics.

The following upgrades were implemented in the game:

**Enhancer 1:** Increases the potency of all eligible unlocked abilities by 30%! (e.g. All Poison Damage)

**Enhancer 2:** Abilities that can be detonated can now be detonated an additional time!

**Enhancer 3:** The Size and Lifetime of your exploration Senses will get refilled when discovering a resource!

**Enlarger 1:** Increases the size of all eligible unlocked abilities by 30%! (e.g. Poison Spores / Poison Trail)

**Enlarger 2:** All new Connections your Mycelium Network will generate are 30% wider!

**Enlarger 3:** Your exploration senses will be 30% bigger and decay 30% slower outside your network!

**Poison 1:** Learn how to send out Poison Spores from the Base Center to find and detonate near predators to strongly poison them!

**Poison 2:** Mutate your Mycelium Network to weakly poison all predators who feed on it!

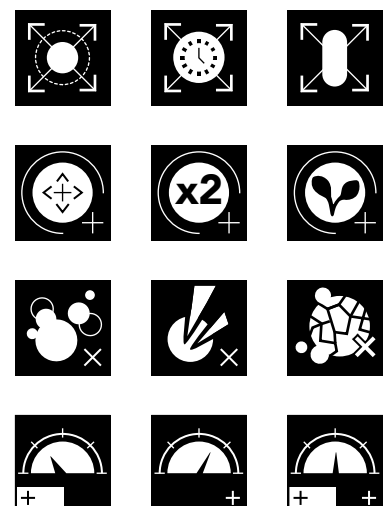
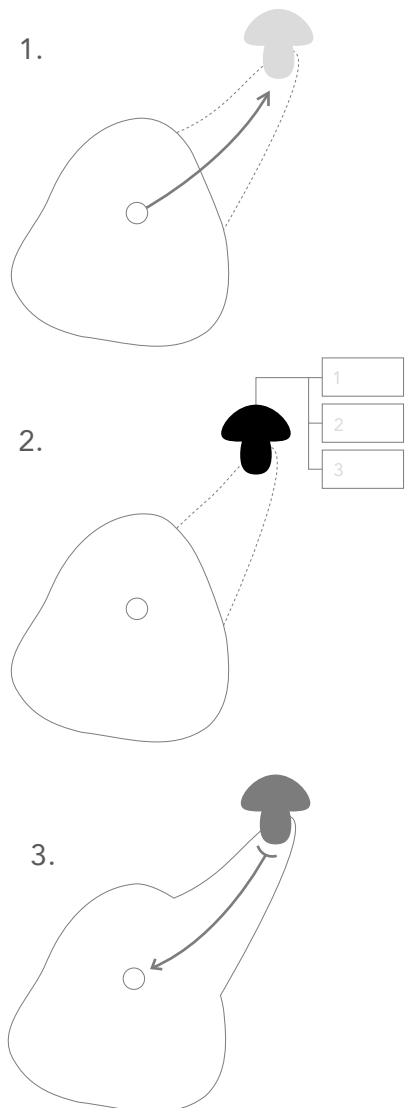
**Poison 3:** Leave a small poison trail behind you when moving around in order to moderately poison predators!

**Speed 1:** Move 30% faster when exploring outside of your Mycelium Network!

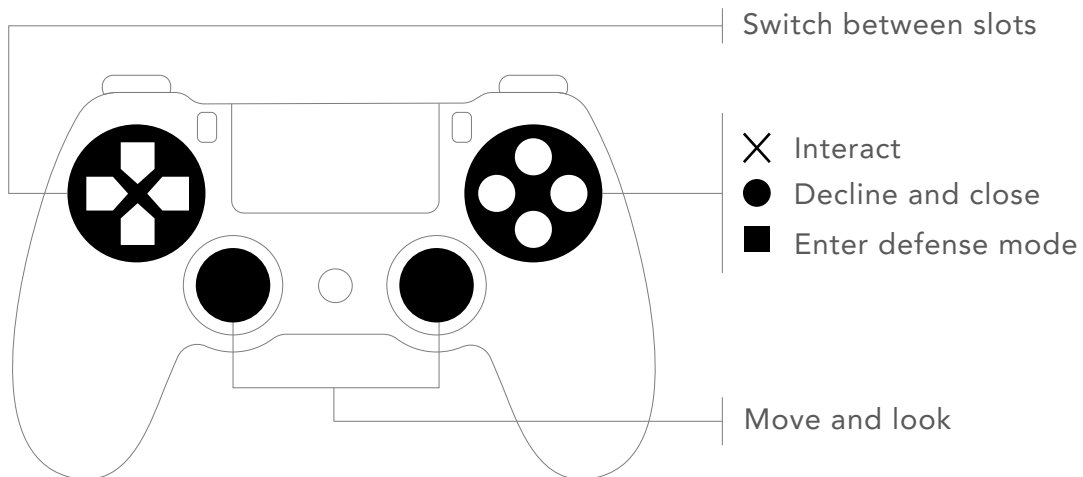
**Speed 2:** Move 30% faster when moving on your Mycelium Network!

Always move 20% faster!

**Speed 3:** Always move 20% faster!



## Controls



## Prioritized Learning

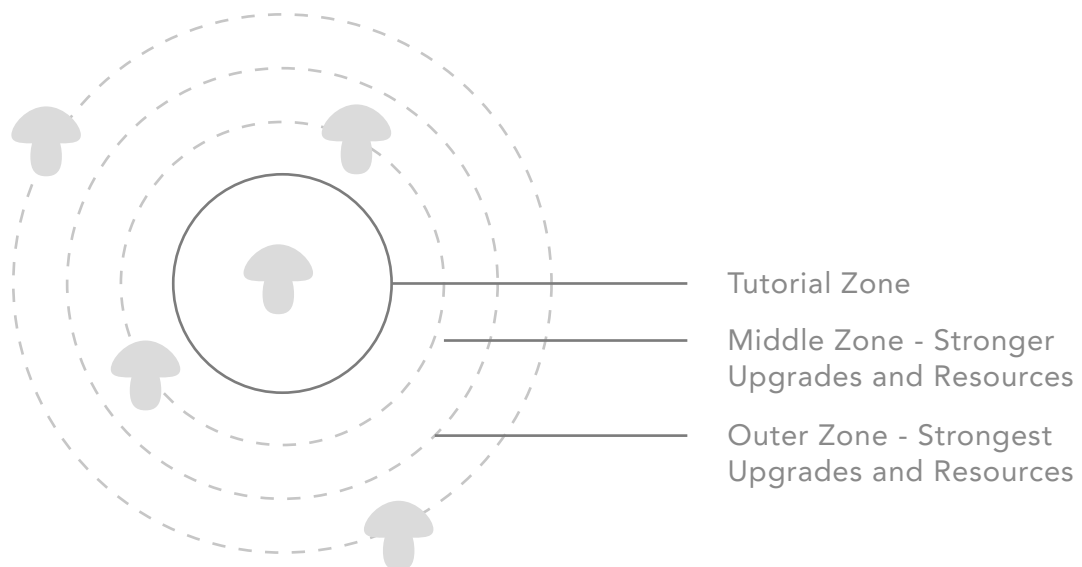
A	B	C	D	E
Prio	Feature to teach	Difficulty	Tutorial Order	Onboarding Plan
1	Moving	0	1	Showing the controls in the beginning of the game
1	Connecting to Resources	1	2	After the players confirm the controls they get a small popup telling them to move to a plant and have the mycelium connect to it.
3	Get Resource Information	1	3	When moving close to the Resource node the UI button pops up prompting them to inspect the plant further.
1	Player losing connection to base	3	-	A warning sound and a popup tell the player they have been disconnected and should move back to the mycelium to reconnect
1	Connecting to Upgrades	1	4	The tutorial popup will tell the player to move to an upgrade fungus after connecting the resource plant.
1	Selecting Upgrades	1	5	The tutorial popup will tell the player to interact with it with the X Button and prompt them to select an upgrade.
1	Enabling Upgrades in Base	2	6	The tutorial popup will then tell the player to move to the base center and activate the newly acquired upgrade.
2	Resource Costs	2	7	When the players go to the base to activate the upgrade they will be met with the base UI.
1	Base Resource Cost	3	8	When activating the upgrade the UI will visualize the additionally allocated resources
3	Enemy Timer	2	9	The base UI also visualizes the resource inflow, the current usage and the remaining resources
1	Enemies are present	2	10	When activating the poison upgrade the enemy timer will be shown at the top of the screen.
2	Combat	2	11	When the enemy is present it will start blinking to warn the players further. They will also appear on the players compass.
2	Losing Health	3	-	Depending on which upgrade they chose this is easier or harder to grasp. The most difficult one being the poison spores.
				If they move to the base center now a square UI popup appears telling them to enter the base defense. From there on a UI is guiding them through the process.
				When the players start losing health a bar will pop up showing it slowly decreasing as well as a warning sound starts playing.
				When entering the base UI they will see what the problem is based on the resource bars.

A	B	C	D	E
Prio	Feature to teach	Difficulty	Tutorial Order	Onbo
1	Moving	0	1	Showi
1	Connecting to Resources	1	2	After t
3	Get Resource Information	1	3	When
1	Player losing connection to base	3	-	A warn
1	Connecting to Upgrades	1	4	The tu
1	Selecting Upgrades	1	5	The tu
1	Enabling Upgrades in Base	2	6	The tu
				When
2	Resource Costs	2	7	When
1	Base Resource Cost	3	8	The ba
3	Enemy Timer	2	9	When
1	Enemies are present	2	10	When
2	Combat	2	11	Deper
				If they
				When
2	Losing Health	3	-	When

E				
Onboarding Plan				
Showing the controls in the beginning of the game				
After the players confirm the controls they get a small popup telling them to move to a plant and have the mycelium connect to it.				
When moving close to the Resource node the UI button pops up prompting them to inspect the plant further.				
A warning sound and a popup tell the player they have been disconnected and should move back to the mycelium to reconnect				
The tutorial popup will tell the player to move to an upgrade fungus after connecting the resource plant.				
The tutorial popup will tell the player to interact with it with the X Button and prompt them to select an upgrade.				
The tutorial popup will then tell the player to move to the base center and activate the newly acquired upgrade.				
When the players go to the base to activate the upgrade they will be met with the base UI.				
When activating the upgrade the UI will visualize the additionally allocated resources				
The base UI also visualizes the resource inflow, the current usage and the remaining resources				
When activating the poison upgrade the enemy timer will be shown at the top of the screen.				
When the enemy is present it will start blinking to warn the players further. They will also appear on the players compass.				
Depending on which upgrade they chose this is easier or harder to grasp. The most difficult one being the poison spores.				
If they move to the base center now a square UI popup appears telling them to enter the base defense. From there on a UI is guiding them through the process.				
When the players start losing health a bar will pop up showing it slowly decreasing as well as a warning sound starts playing.				
When entering the base UI they will see what the problem is based on the resource bars.				

## Level Design

The level is designed to help players understand its mechanics from the beginning, providing them with the crucial upgrade needed for defense. As the game progresses and players expand their network into uncharted territory, they will come across resource nodes that offer increasingly valuable and rare resources. Furthermore, they will encounter better upgrade nodes that are essential for survival in the growing challenges of the environment. The level design supports this progression, allowing players to grasp the game's mechanics while gradually introducing them to more advanced resources and upgrades.



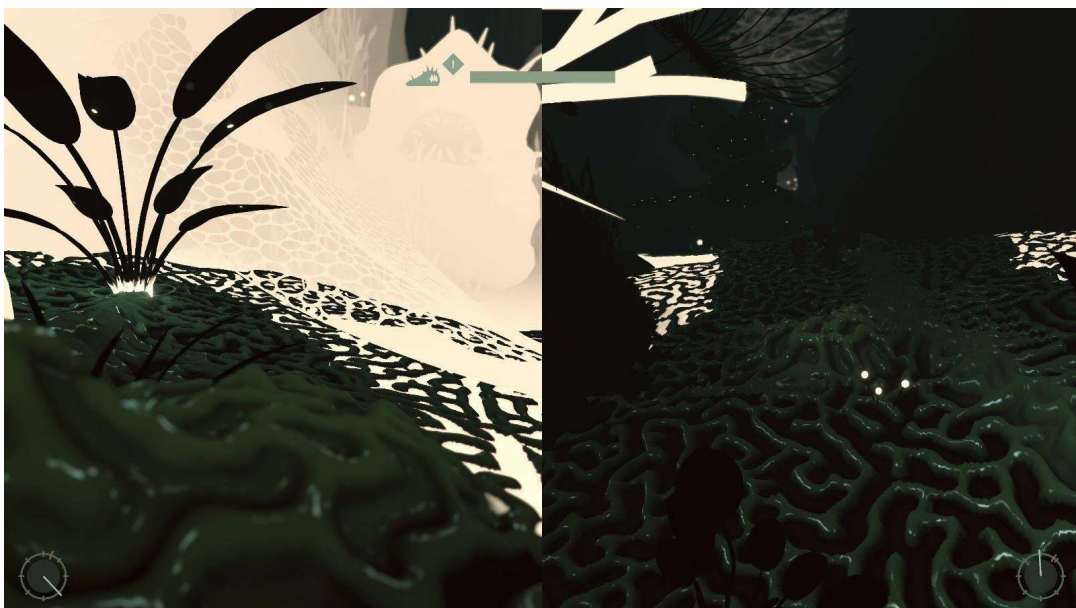


## Visuals

The visuals aim to immerse the player in the perspective of a mycelium organism. The environment should evoke a sense of mystery, strangeness, and unfamiliarity. The art style must effectively convey the alien-like abilities and lifestyle of fungi. This effect should be created by our color scheme, the plants that are moving and shaking as the players get closer, the size of the objects, indicating that the players are small, and the player-characters design itself which consists of a wobbling hill inside the mycelium which is spreading its goo over the floor as it leaves the base.

Given the abstract nature of our concept, clear visual communication becomes crucial. We strive to strike a balance between familiar elements that players already know and introducing new aspects to create an enjoyable yet slightly unsettling experience.

To aid players in navigating the game, we employ a clear color coding system. Darker elements represent those elements that belong to or are meant for the player, while the non-interactive outside world remains white. As the player connects with plants, the surrounding nature gradually turns black, symbolizing the symbiotic partnership between plants and fungi.



## Sound

The music and sound effects in the game were designed to create a blend of surreal synth and natural elements. This combination aimed to represent both the fungi's connection to nature and their intricate growth, as well as evoke a sense of surrealism.

The influence of synthesizer sounds in the music was also inspired by the intriguing fact that fungi can produce musical sounds when connected to a synthesizer. This discovery further reinforced the vision for the game's audio, tying it closely to the theme of fungi and their unique qualities.

## Music

The music in the game is designed to have three distinct states that correspond to different game-play situations.

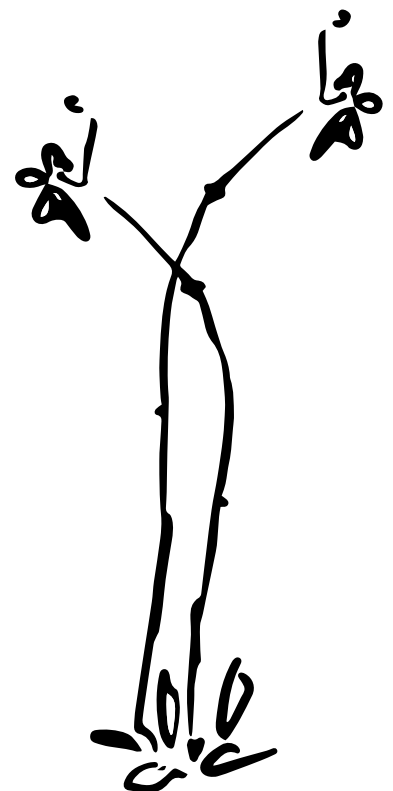
The first state occurs when players are on their mycelium network, creating a sense of comfort and safety. The music in this state aims to make players feel at home and at ease within their familiar territory.

In the second state, when players venture outside their network and explore the unknown, the ambience shifts to a more tense atmosphere.

Chattering sounds of insects and plants are added to the music, enhancing the sense of exploration and heightening the feeling of being in unfamiliar territory.

The third state of the music occurs during enemy attacks. The music becomes more intense, creating an uncomfortable and alarming sensation. This heightened musical state serves as a warning to players, alerting them to the imminent danger and urging them to take defensive action.

Furthermore, as the players' mycelium network decays and they begin to lose the game, the music intensifies even further. This intensified version of the music serves as a powerful indicator of the impending demise, adding urgency and tension to the gameplay experience.



## Sound effects

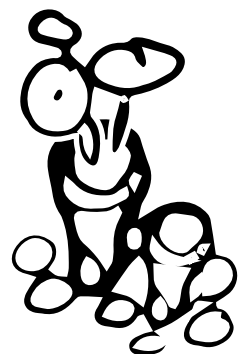
The sound design in the game combines both realistic and synthesized sounds to enhance the overall experience. The organic creatures, with their movement and behaviors, are represented by more natural and realistic sounds, creating a grounded and immersive atmosphere. These sounds aim to evoke a sense of being in a natural environment and interacting with living organisms.

On the other hand, the surrounding elements and the user interface (UI) incorporate synthesized sounds to create a surreal and otherworldly ambience. These synthesized sounds add a touch of abstraction and uniqueness to the game, reinforcing the surreal feeling associated with the complex growth of the mycelium network. They contribute to the overall audio aesthetic, creating a distinctive and immersive audio landscape.

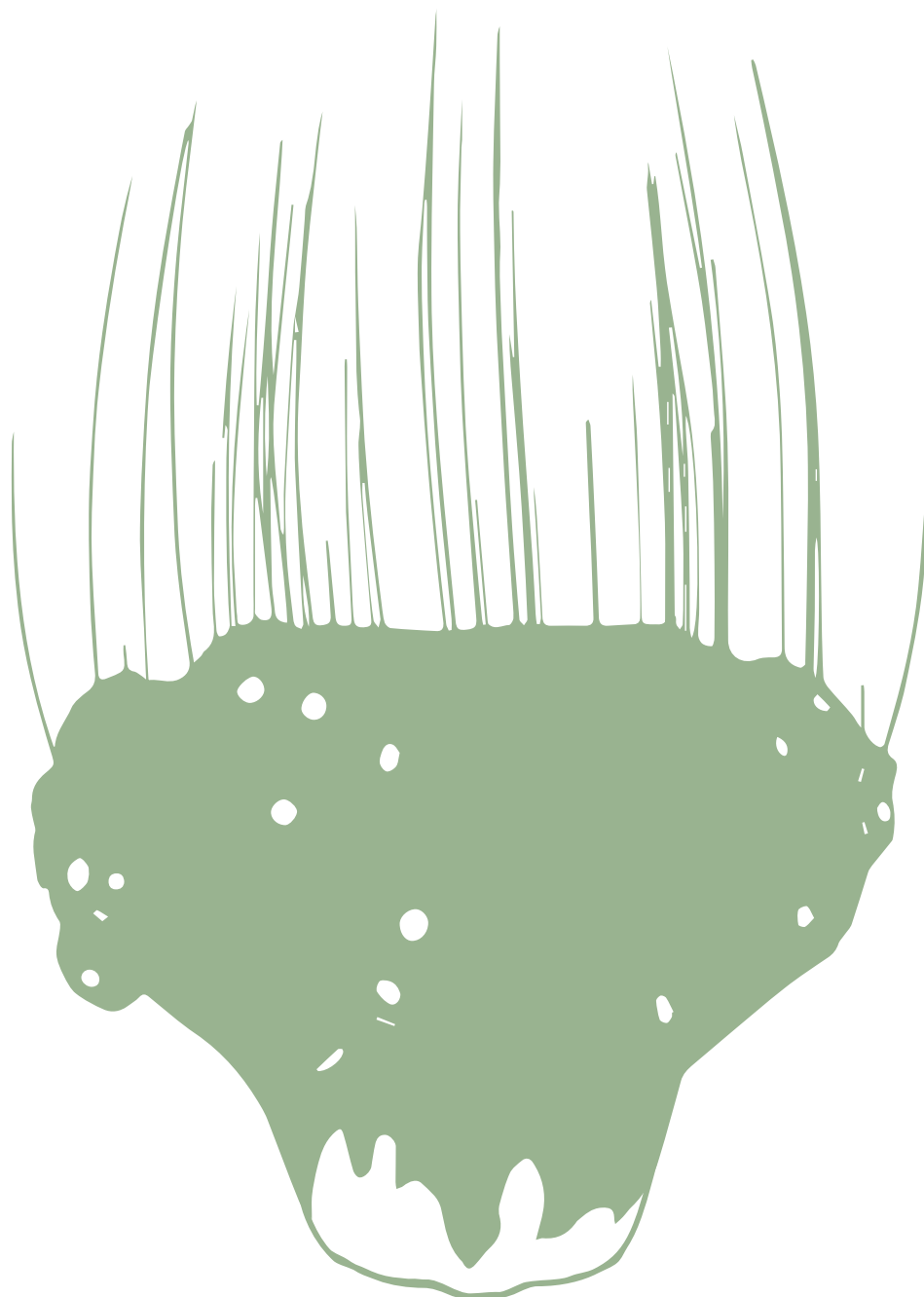
By blending realistic and synthesized sounds, the game achieves a balance between familiarity and otherworldliness, immersing players in an atmospheric audio experience that complements the visuals and gameplay mechanics.

## Cooperation

The entire music composition and sound effects were skillfully created and implemented in FMOD by Reina Burkhalter. Her work not only met but exceeded our expectations, delivering a phenomenal soundscape that perfectly captures the intended atmosphere of the game.



## 6.0 CONCLUSIONS AND OUTLOOK



# 6.1 Conclusions and Outlook

## ◦ Conclusions

*Trying to capture the essence of how fungi act and what defines them as an organism was an extremely challenging task, which revealed the deep interconnection of our initial questions.*

As outlined in the documentation, the most challenging part was to give the network a natural feel, making it behave as if it were alive and allowing players to play a small part in shaping the mycelium as a whole. The solution we found was to have the network operate autonomously, similar to an NPC.

Fungi exhibit optimized behavior, forming connections that resemble human engineering, making it difficult to convey a sense of them being living organisms, as their solutions appear highly logical. With this understanding, we provided players with a tool to act as the senses of the network. When a player discovers something, the network will attempt to find the most efficient way to connect to it, utilizing the information provided by the players.

These connections had to continuously form in an organic manner to reinforce the impression of a living creature.

The concept of a shared overarching entity allowed us to use it as a focal point for communication between the players. All resources were shared between the players and the network, necessitating their allocation as a team. This ensured the mycelium had enough to survive while the remaining resources were invested in the best upgrades to expand and tend to the mycelium network optimally.

These upgrades can be activated and deactivated by both players for both players. This requires teamwork, as players can assist one another by activating an upgrade for a player who is far away from the center of the network. Additionally, players can specialize in specific areas of the game through these upgrades, depending on their preferred playstyle. Finding the most optimal survival solution opens up further avenues for discussion among the players.



The upgrades are obtained by forming symbiotic relationships with other fungi in the natural environment surrounding the mycelium. By providing these fungi with the required resources, they, in turn, offer players various types of improvements that can be applied to their player characters or mycelium network. This includes enhancements to player stats, how the network connects to resources, or how other upgrades interact with the player. Defensive abilities are also included, as they are necessary to fend off invading predators that attempt to feed on your mycelium, disconnecting it from its resources and causing it to starve.



## ◦ Challenges & Learnings

### Challenges

We set out with the goal of creating a project that would require teamwork, and that certainly held true for the final product. Establishing a strong connection between tech and art was crucial in order to accomplish our objectives. It presented challenges, but the rewards were equally satisfying. Communication flowed smoothly, and we genuinely enjoyed working together as a team.

"Multifarious" was an ambitious project with a broad scope. Overall, we successfully achieved what we had set out to do, although we had to simplify certain features and make some cuts. For instance, there are fewer enemies and upgrades than we had originally planned.

### Learnings

This project provided us with numerous opportunities to explore new topics that we had not previously delved into. We extensively researched fungi and their significant impact on the world. Additionally, we acquired knowledge about new shader and particle system techniques, delved into asynchronous calculations, and had the chance to work with render textures.





## ◦ Outlook

The prototype we were able to develop within the given time provides a glimpse of what the game could become if expanded upon. It demonstrates how players can search for resources, acquire and combine upgrades, and face challenges from enemies that threaten their mycelium network.

To transform this prototype into a fully developed game, we would need to expand on all the existing features in some way. One of the key areas of focus would be the initial player experience, as newcomers should be guided and introduced to the game gradually, considering the array of unfamiliar and complex features, especially if the game were to become even larger in scope.

The playable map would need to be expanded, placing greater emphasis on exploration as an integral part of the gameplay experience. Currently, players do not have to venture far from the base to find everything they need, and the risk associated with leaving the network for an extended period of time is relatively low.

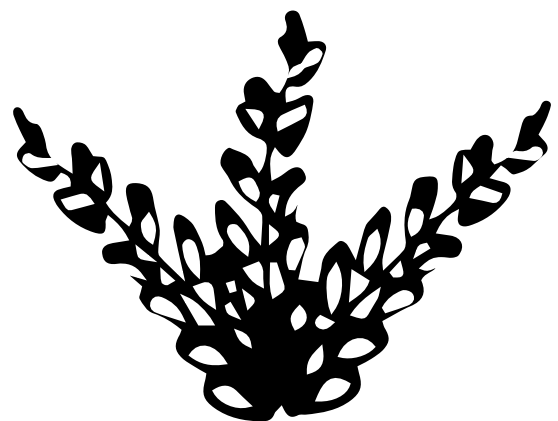
A wider variety of upgrades would need to be introduced. The current combinable and stackable nature of upgrades is interesting, allowing for multiple interactions and the potential for significant effects. By incorporating more, particularly stronger or more impactful upgrades, players would have a greater incentive to seek out different combinations and discover new possibilities in each playthrough. The distribution of upgrades and resources should also be randomized in the final game.

The addition of new tools for players would only enhance the gameplay if they were essential in overcoming specific challenges. Therefore, incorporating additional enemy types that behave differently and require creative problem-solving by the players becomes crucial in the final product.

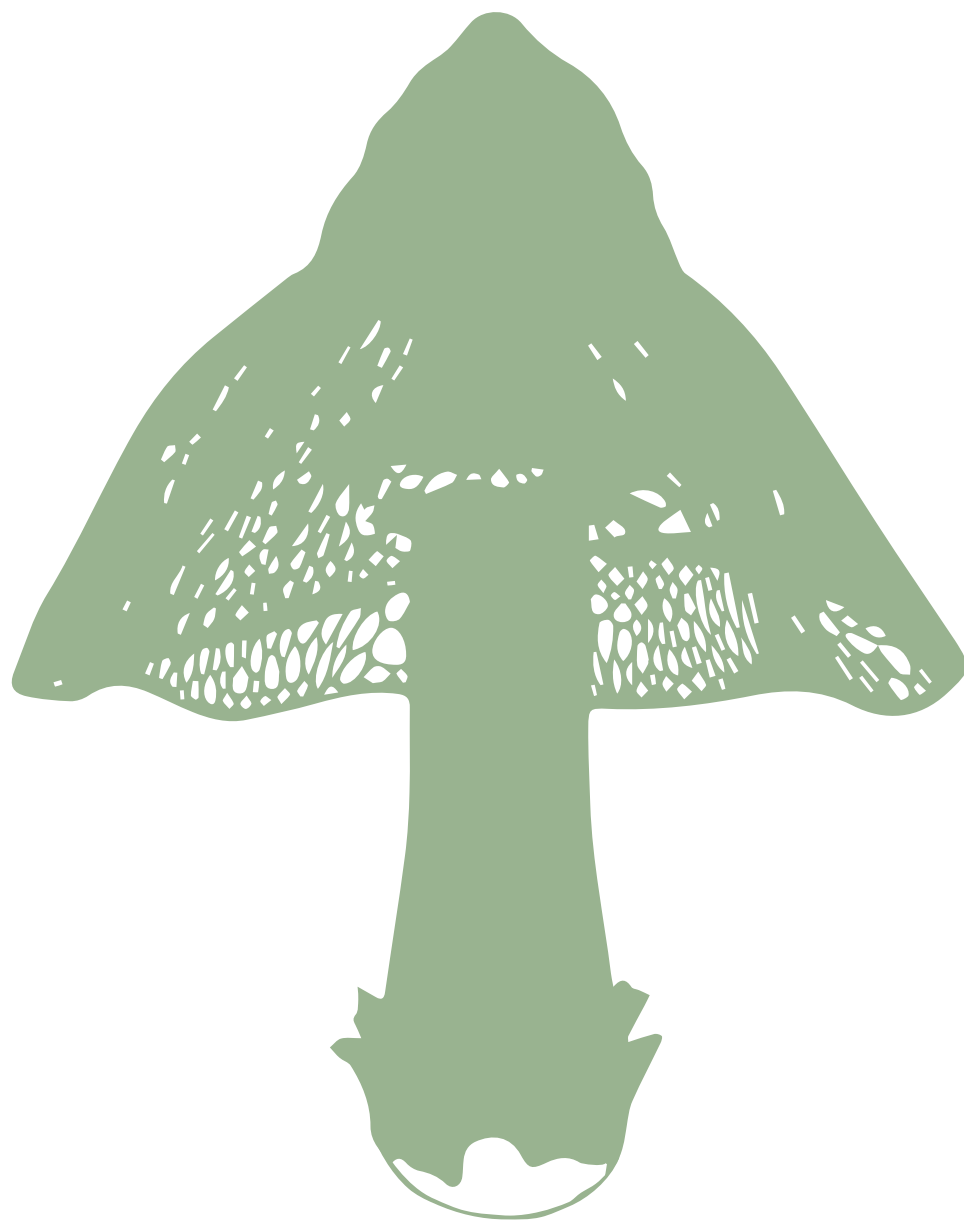
By addressing these points, we are confident that the game could evolve into a fully functioning, replayable co-op survival roguelike experience.



However, it's important to note that the game primarily served as a portfolio project for both of us, allowing us to challenge ourselves in creating a unique game while leveraging the various tools at our disposal. The end result displays our skills and will certainly be uploaded to itch.io and showcased in our respective portfolios.



## 7.0 SOURCES



# 7.1 Sources

## IMAGES

[Figure 1] Nobody Saves the World, DrinkBox Studios, <https://news.xbox.com/en-us/2021/08/26/nobody-saves-the-world-launching-early-2022-with-online-co-op/>

[Figure 2] <https://waytoomany.games/2022/01/18/review-nobody-saves-the-world/nobody-saves-the-world-5/>

[Figure 3] Minecraft, Microsoft, <https://www.gadgets360.com/internet/news/security-flaw-software-bug-internet-web-server-minecraft-2646511>

[Figure 4] Minecraft, Microsoft, <https://www.indiatimes.com/technology/gaming/indian-gaming-creators-minecraft-youtube-557901.html>

[Figure 5] Portal 2, Valve, <https://steamcommunity.com/sharedfiles/filedetails/?id=2931451627>

[Figure 6] It takes two, Electronic Arts, <https://news.ea.com/press-releases/press-releases-details/2021/It-Takes-Two-EA-and-Hazelights-Thrilling-Co-op-Only-Action-Adventure-Game-Launches-Today/default.aspx>

[Figure 7] Dome keeper, Raw Fury, <https://www.pcgamer.com/dome-keeper-review/>

[Figure 8] Dome keeper, Raw Fury, [https://www.reddit.com/r/PixelArt/comments/xoli44/procedural\\_palette\\_swap\\_in\\_dome\\_keeper/](https://www.reddit.com/r/PixelArt/comments/xoli44/procedural_palette_swap_in_dome_keeper/)

[Figure 9] Dome keeper, Raw Fury, <https://www.shacknews.com/article/132448/-dome-keeper-review>

## TOOLS

### Unity

Visual Studio Community

Photoshop

Github Desktop

Gitlab

FMOD

### Unity Packages:

DoTween

Cinemachine

Kawase Blur

### Organisation:

Discord

Notion

Google Drive

Blender

Illustrator

Procreate

Houdini

`blender-differential-growth > inca` <https://github.com/inca/blender-differential-growth/releases/tag/2-1-1>