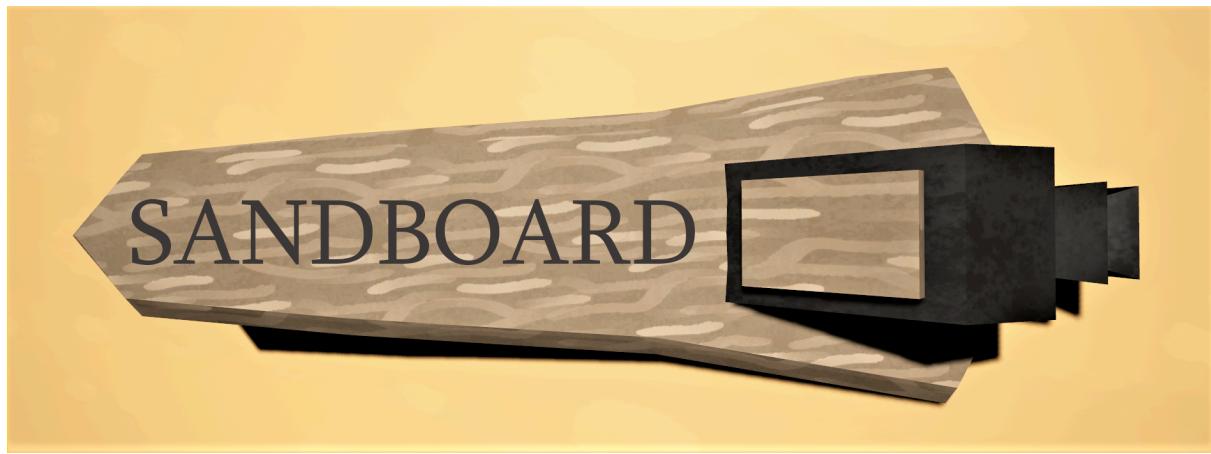


Executive Summary



Sandboard logo

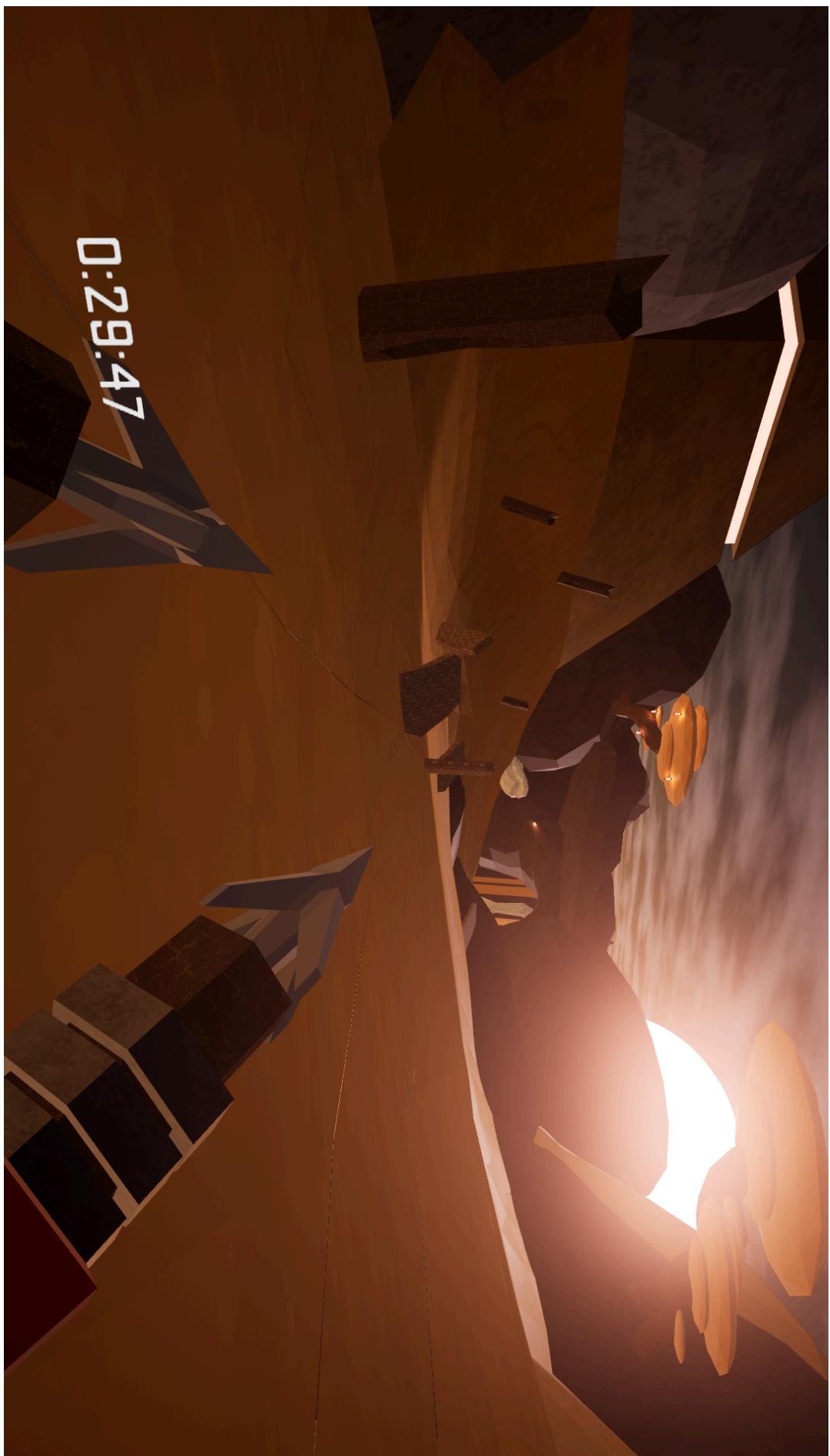
Sandboard is a VR dune surfing and grappling game made in Unreal Engine 4 for Oculus Quest/Quest 2. This game has an emphasis on playability in a small, limited mobility space while still giving the player a sense of speed and maneuverability. With a board at their feet and two grapple hands at their disposal, players zip down sandy slopes, jump off planks, rocks, and boosts, and use their grapples to propel themselves forward and upward.





In-game screenshots

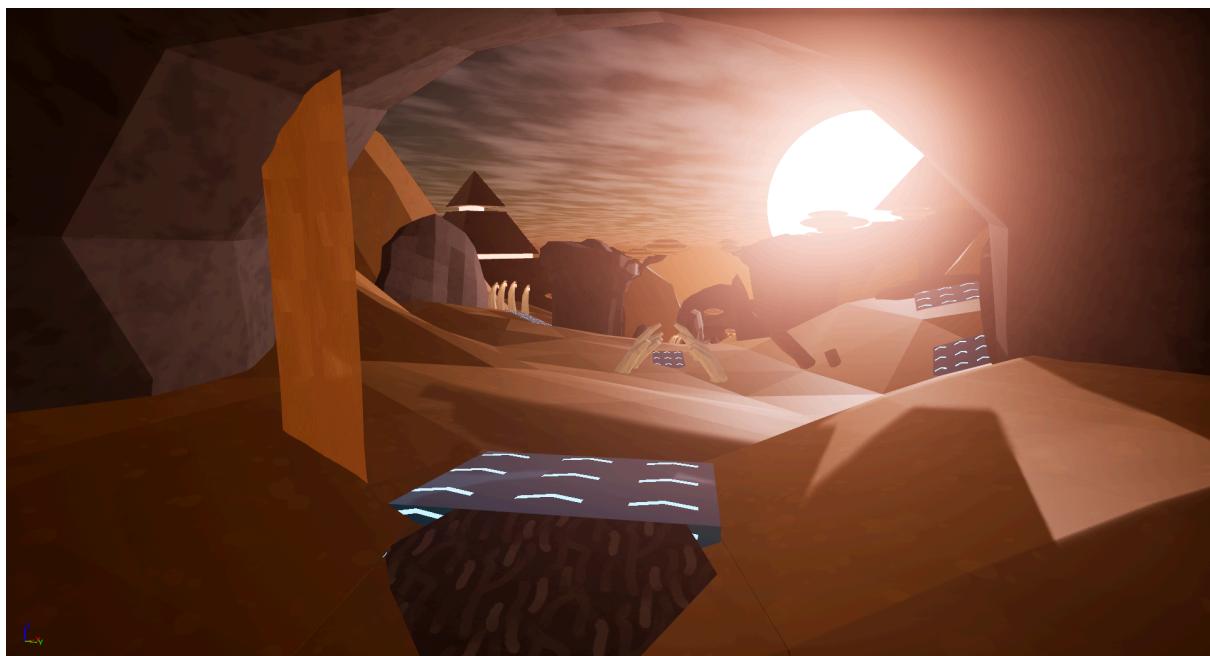
Due to the lack of VR surfing games, this game offers a new type of VR experience. While this type of gameplay is somewhat new to VR, players may recognize similar elements from non-VR games such as shield surfing in *The Legend of Zelda: Breath of the Wild*. Grappling and swinging is always a popular VR gameplay mechanic and is used in many other VR games, adding a bit of familiarity from the VR world. Ultimately, while this game brings something new to VR, it still feels familiar enough to play without much effort.



Prototype Report

Gameplay

The goal of the game is to reach the end of the level as fast as possible by moving through the level. Your current time is displayed in the bottom left corner. There are two primary forms of movement: surfing and grappling.



Boosts

Anytime you are grounded, you surf in the direction you are looking. You will move faster downhill and slower uphill. In-game, you will encounter “boosts”: blue jumps or panels with moving chevrons. Surfing on these will boost your speed. Jump boosts will keep your velocity as you fly off, increasing your speed and propelling you great distances.

By squeezing the controller grip, you will fire a grapple from the corresponding hand. If the grapple hits something, it will latch on and pull you there. Grapples can latch onto almost anything, even from far away. You can use both grapples at once to move faster and even lift off the ground. Releasing the grip or passing by the grapple will retrieve the grapple, at which point you can use it again. Grappling is very useful for speeding up, getting the most out of jumps, and avoiding obstacles.

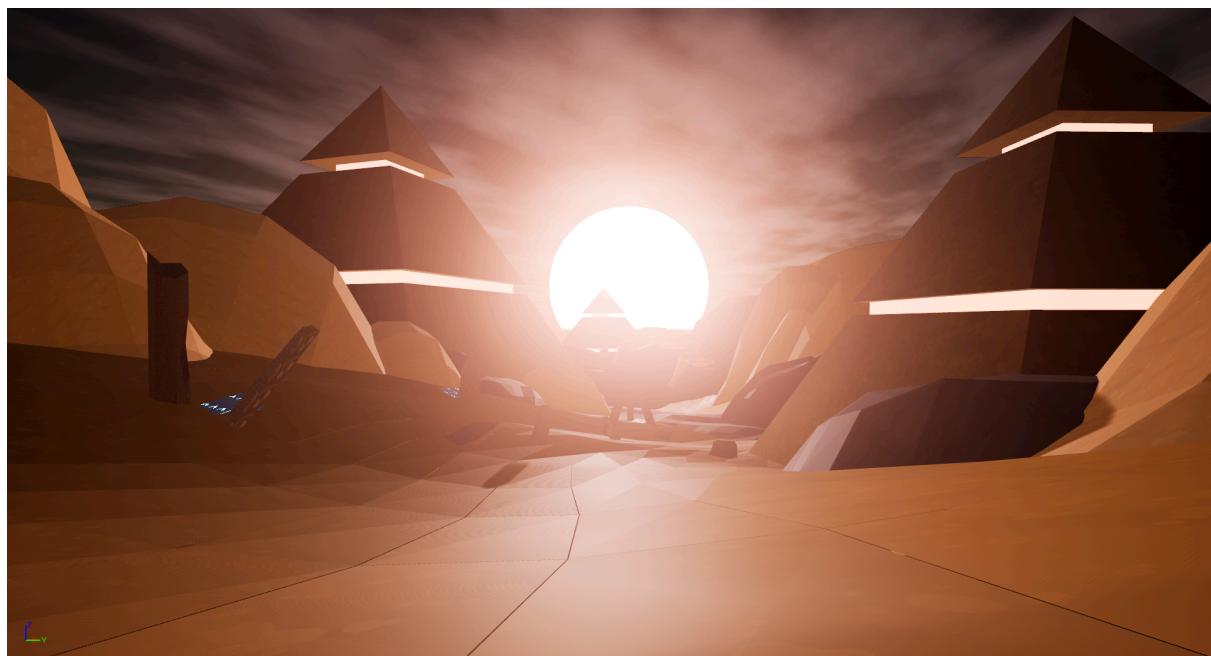
Some turns can be hard to make by whipping your body around. If you would rather not rotate your whole body, you can snap-turn your view with the left joystick. This can be useful when you stray from your intended path or hit an obstacle.

Levels



Level selection

From the menu, you can select a level to play. Levels in *Sandbox* consist of different areas, each testing different movement skills, and take about a minute to complete on an ideal playthrough. As you near the end of a level, you will see a bright red panel on the ground. Reaching this panel will complete the level, display your final time, and take you back to the menu. Levels feature multiple paths, meaning players can try out different paths to find the fastest way through. The game includes two levels.

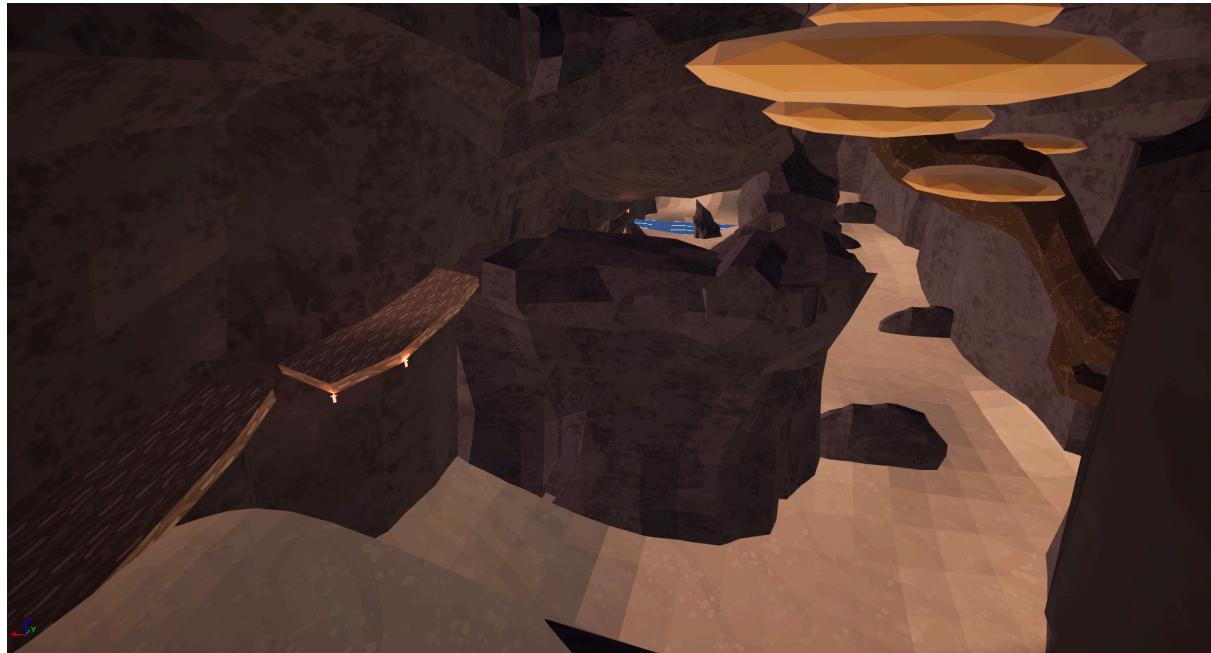




Arches

The first level, Arches, is relatively smooth, easygoing, and good for getting accustomed to the movement and controls. It features a balance of sloped, linear paths with obstacles and open areas where players have more freedom of movement. The scale of the pyramids and dunes on this map make for easy grappling across wide areas while grappling to the stone arches above can get you past obstacles.





Ravine Town

Ravine Town, the second level, is much more claustrophobic and difficult. This level's tight turns, cramped tunnels, and thin bridges test the player's movement abilities. However, players who notice the path they take goes above an earlier segment can use this information to grapple up and skip the toughest part of the level.

Development Report

Concept

The game was initially inspired by driving a buggy around the desert with my grandfather and watching motorcyclists speed along the desert trails. I connected this with sand surfing in games such as *The Legend of Zelda: Breath of the Wild*'s sand seal shield surfing and *Journey*'s sand surfing scene. The VR aspect of the game inspired a front-facing, hands-on form of surfing akin to skiing. Rather than skis, however, I decided to implement grapples similar to those in *Swarm* and *Spider-Man: Far From Home Virtual Reality* to allow for more vertical and personalized movement.

I used this idea for my DIM project primarily because I am unaware of any game quite like it currently available for VR. The closest game I can think of is *Swarm* due to its grapple-based movement, which was the main inspiration for *Sandboard*'s grappling. I also chose this idea because I wanted more experience working with VR as I plan to do more VR development in the future.

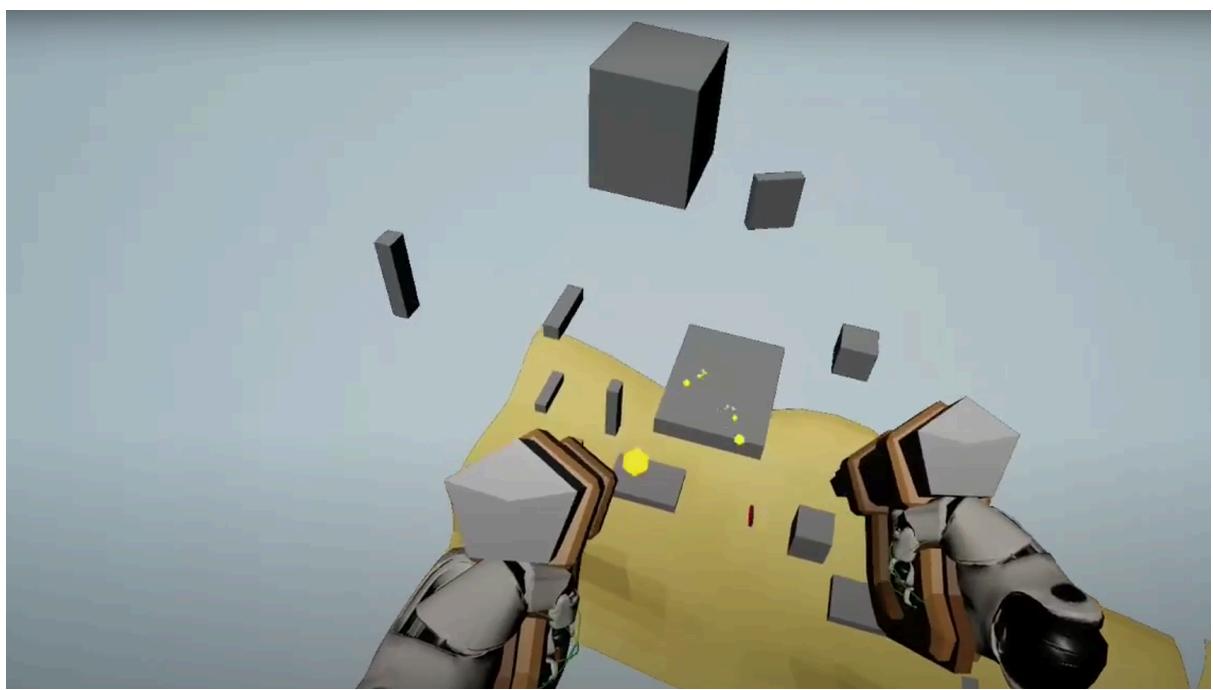


Early stages of surfing and grappling development

I have had the idea of a sand surfing game for a few years but only recently realized its potential in VR. While the limitations of the Quest prevented me from making a more open world speed-focused game like *The Pathless*, designing more linear levels worked out in the end since I was able to put more detail into a more compact space. This also helped me focus the game on maintaining speed and getting to the end of the level as fast as possible.

Development and Changes

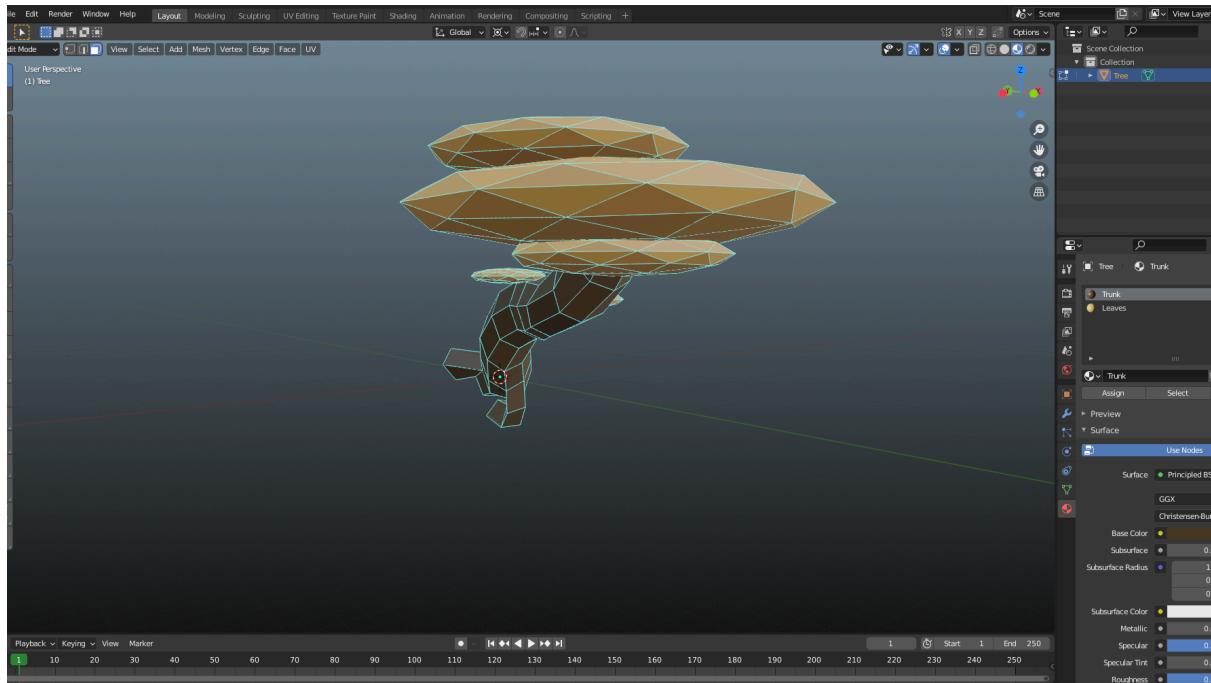
The biggest change to this project was the omission of combat. I planned to pair the movement system with a simple shooting combat system where enemies would surf, swim, or fly beside you and slow you down if not dealt with. One of the biggest facets of this project's identity was to be massive boss fights at the end of each level similar to those in *Shadow of the Colossus*. These would be a test of both the player's movement and combat abilities. In the end, however, the addition of combat could not be justified by the amount of work required to create animations, AI, and areas for these enemies.



The pistol used for combat in early development

The lack of combat may be for the best. Prioritizing movement and level design has allowed me to further develop these areas and strengthen the core mechanics of the game. Combat, while appealing, would have detrimental effects on players easily overwhelmed by the need to navigate, grapple, and shoot all at once.

Assets



A tree mesh in Blender

I made almost all of the assets for this project. I created 3D models in Blender, textures in Gimp, and sound effects and music in Logic Pro. The font was taken from Google Fonts and the sun and clouds are the default UE4 skybox and directional light.

The music was one of the first aspects of the game I worked on. As my ideas about the visual side of the game were still forming early on, I decided creating some concept tracks would help me formulate an artistic direction. Most of what I produced was very percussive with a high-pitched melody that would only be present half the time. I used reverb, delay, and more sampler instruments than usual as opposed to synths. This was to make the tracks as natural sounding as possible while keeping a sense of urgency through the percussion and bass.

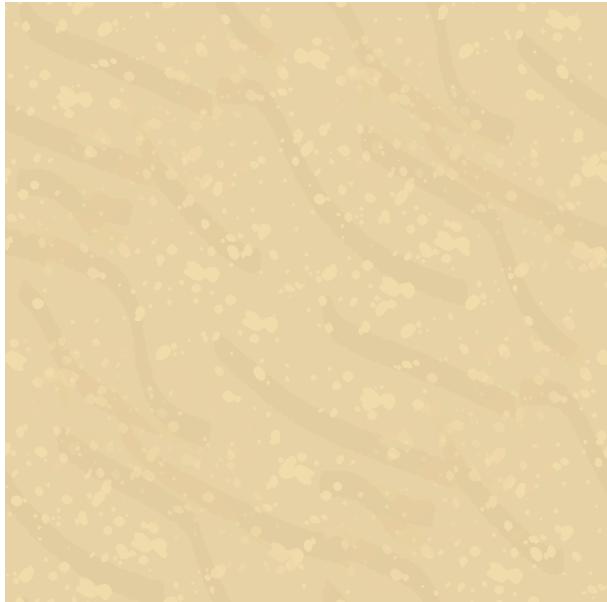
Over time, the music I produced got calmer and more natural-sounding to match the aesthetic of the flowing sand and open areas of Arches. Despite my continual work on the project's music direction, two of the first three songs I wrote ended up being the best fits for the levels I made and are now the music for Arches and Ravine Town.

The making of the sound effects was relatively straightforward. The grapple noises were created using a trap drumkit and some basic effects like compression and distortion to make the sound feel heavy and cohesive. My favorite part of these sounds are the hi-hats. I changed their amplitude and pitch over the course of the sound effect to simulate a chain or propulsion device rolling inside the grapple hands. The surfing, menu wind, and grapple retrieval noises were created in the Massive synth using white noise with phasers and slight pitch bend. I also EQ'd all the sound effects before finishing to make them sound clean and each take up their own frequencies so as not to overlap with one another. However, due to the fact that I used similar drums in the music and sound effects, the music can make some of the sounds less prominent.



Collection of 3D models used in the menu

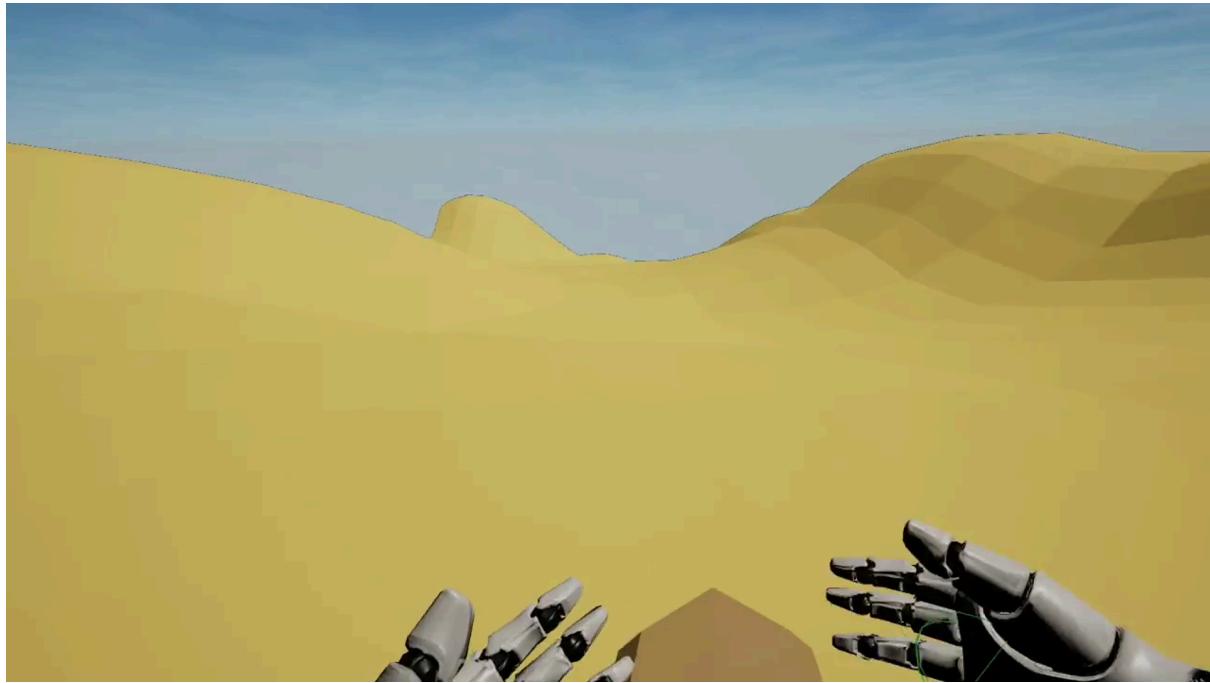
I made all the 3D models low-poly for a few reasons. First and foremost, I knew I would develop this project for Quest, meaning the graphics had to be low end. However, as I came to realize I would have trouble running this game on standalone Quest and would probably need to wire it to a PC, I improved the lighting and began using more meshes for detailing. I also used a low-poly art style because I have experience with it; I have very little experience with modelling high-poly meshes and figured it would be safest to stick to what I know. Another benefit of low-poly art is the ability to generate new assets very quickly, giving more time to the overarching level design process.



Sand texture with a base color, a layer of darker lines, and a layer of lighter flecks

The textures for this project were made in Gimp. Most textures consist of a base color and one or two textures with variations of the same color layered on top. The textures are used in UE4 as materials, where effects such as tinting, tiling, and UV panning could be applied. Once I made base textures like wood, metal, and rock, I could easily make new meshes without having to worry about texturing most of the time. Additionally, simple meshes like planks and bones could be stuck together inside of UE4 as a new mesh, meaning I did not have to jump between Blender and UE4 every time I wanted a new variation of a mesh.

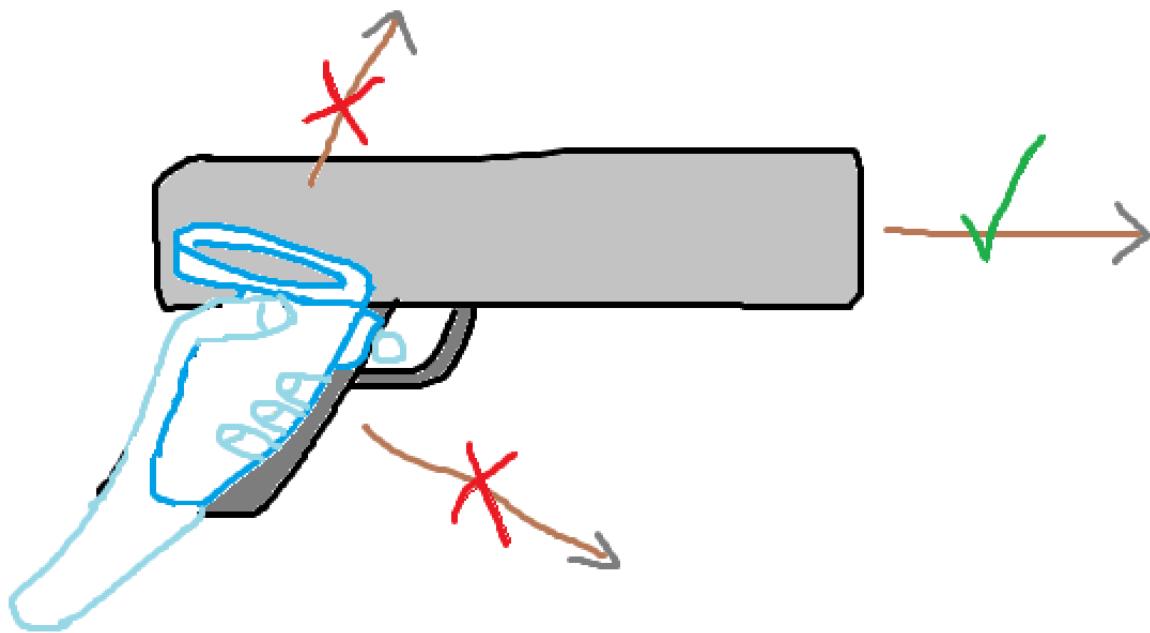
Code



Earliest implementation of surfing

Despite this being my first time completing a VR project, the trickiest bits of code were actually the surfing and grappling mechanics. I coded the grapples to work like grapples in Respawn Entertainment's games, meaning that they can be used to bypass points grappled to by providing additional input and are more useful as tools of propulsion rather than redirection. However, grapples can still be used to reach desired areas.

Most of the time spent on code was focused on tweaking and balancing these features to achieve a natural-feeling movement system. This was the toughest part of the project because the movement is so core to the gameplay that any small alteration could change the whole feel of the game and potentially ruin whole areas of levels. With this in mind, I put off level design until I was comfortable with the state of the movement system.



When trying to grapple, players expect a bit of an offset

I especially had trouble with the direction in which grapples fired. It feels unnatural for a grapple to fire directly away from your controller, which I eventually realized is due to how we hold handles. When holding an object such as a gun, the handle is at an angle, not straight up and down. This angle took some work to find, but I eventually found a natural-feeling offset from which to position the grapple hands and grapple firing origin point.

I used the UE4 VR Expansion Plugin for this project. While UE4 manages many aspects of VR development such as head movement and controller tracking by default, the VR Expansion Plugin included an accurate hitbox depending on the player's height and a class for grabbable objects. However, much of what I used it for is no longer in the game, such as pistols and climbing.

Level Design

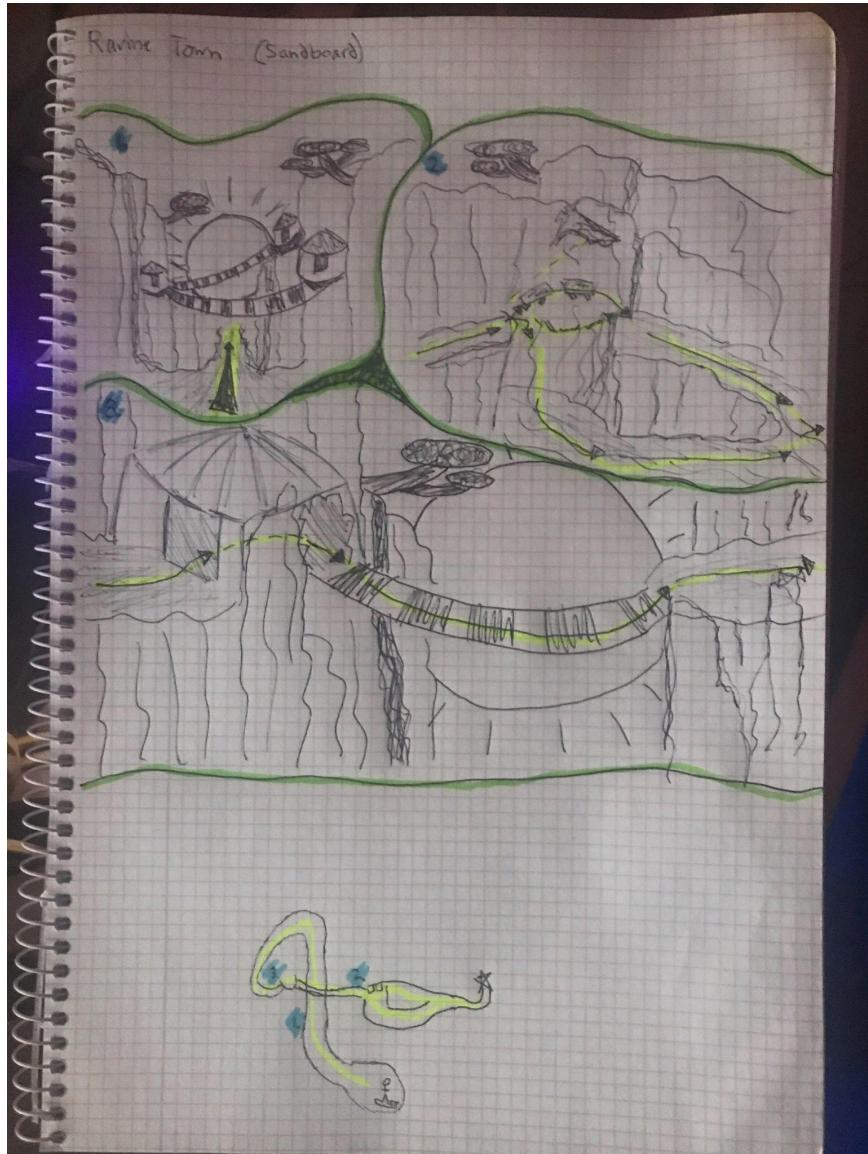
Outside of movement and VR technicalities, coding this project was fairly simple.

Once I had roughly finished the movement system, I moved to level design, which proved to be much more challenging than I expected. The movement system was a huge influence in the level design process; after a new segment was outlined, I would thoroughly test it in VR to ensure it encouraged natural or engaging movement.



Arches from above

Arches was designed to be an entry point to *Sandboard*. By alternating between open, freeform areas and linear ones, I wanted to provide the player with a variety of environments to familiarize themselves with the feel of the movement. I added an obstacle-free runway at the start of the level to give the player a moment to prepare themselves. The next area provides many paths and obstacles for the player to choose from, along which they can try out the grappling hooks. The following areas are varied to allow the player to navigate as they choose.



Sketches of Ravine Town areas

Realizing I only had time to add one more level after how long Arches took to complete, I designed Ravine Town to be much more difficult. As opposed to Arches, this level sends the player down more tight paths. These paths can only be avoided by grappling past them, requiring skill from the player no matter which path they take. Since *Sandboard* has no fail state, I laid out the level so that players who miss jumps or fall off bridges end up in previous areas or slower paths, costing them time. Bridges never take very long to reach from the place below them, meaning fallen players can reattempt to cross them without wasting too much time in already-visited areas.

Reflection

This project was a great entry point to start-to-finish VR development. I learned a lot about world scale in relation to the VR player, conventional VR UI, VR movement techniques, and managing player movement. Level design was unexpectedly difficult. Oftentimes, an action as simple as slightly rotating a jump could result in the player avoiding an obstacle, taking an alternate path, or flying off the level. As I am used to designing levels where players can move in any direction, learning to design linear, unidirectional levels was a challenge. Additionally, the scale of levels was almost overshot, as a player focused on the environment immediately in front of them is unlikely to crane their neck to see the scale of setpieces in the background.

My greatest regret with this project is the lack of combat. Though it may have overcomplicated the otherwise simple gameplay, I would have liked to include the aforementioned boss fights. These would have become a staple of the game and added another level of intensity as well as put the player's surfing and grappling skills to use.

Another regret of mine is the lack of content I resulted with. With the movement system, I have set the groundwork for many different types of levels. I also believe I should have focused more on grappling as a necessary mechanic for some obstacles by including more jumps and hidden areas. I think the inclusion of more purely visual enhancements like passive creatures and people, screen shake, and particle effects would have improved the feel of the game as well.

I am overall satisfied with the final prototype. The levels I created support the player's movement abilities, though Ravine Town was a bit rushed and some of its tight turns could be reworked. Arches turned out to be much better than I expected, primarily because it appeals to multiple skill levels. Grappling, while exploitable, adds depth to the game and is satisfying to use. The movement system is easy to pick up and understand.

I am most proud of the concept and theme of the game. I have wanted to actualize my sand surfing idea for a long time but never found the right place in any of my personal projects. The code and asset creation were areas I have mostly dealt with before, but the idea for the game was something unlike any of my other projects. In combination with grappling, I am very happy with the movement system.

Word count: 2701

References

CreateVR, 2019. *Spider-Man: Far From Home Virtual Reality*. [Steam VR game] Culver City, CA: Sony Pictures Virtual Reality.

Davie. (2021). *Remaking Pathfinder Grapple in Unreal Engine 4*. [Video]. Available at: https://www.youtube.com/watch?v=gcXZ4KzBCP8&ab_channel=Davie [Accessed 25 January 2022].

Epic Games (2014). *Unreal Engine 4*. [Computer program]. Available at <https://www.unrealengine.com/en-US/download> [Accessed January 18 2022].

Giant Squid, 2021. *The Pathless*. [Computer game] West Hollywood, CA: Annapurna Interactive.

Greensky Games, 2021. *Swarm*. [Oculus Quest game] San Francisco, CA: Greensky Games.

Japan Studio & Team Ico, 2005. *Shadow of the Colossus*. [PS2 game] Tokyo, Japan: Sony Computer Entertainment.

Joshua Statzer (2021). *VR Expansion Plugin*. [Unreal Engine 4 plugin]. Available at <https://vreue4.com/> [Accessed January 18 2022].

Nintendo, 2017. *The Legend of Zelda: Breath of the Wild*. [Nintendo Switch game] Redmond, WA: Nintendo of America.

Reids Channel. (2021). *Unreal Engine - Dynamic Grapple Hook Tutorial (1/2)*. [Video]. Available at: https://www.youtube.com/watch?v=TQsg8j-a3Y8&ab_channel=ReidsChannel [Accessed 24 January 2022].

Reids Channel. (2021). *Unreal Engine - Dynamic Grapple Hook Tutorial (2/2)*. [Video]. Available at: https://www.youtube.com/watch?v=PSk9TDpowW8&ab_channel=ReidsChannel [Accessed 24 January 2022].

Reids Channel. (2020). *Unreal Engine - Physics Based Sliding (1/2)*. [Video]. Available at: https://www.youtube.com/watch?v=bXjUE_4-oVo&ab_channel=ReidsChannel [Accessed 20 January 2022].

Reids Channel. (2020). *Unreal Engine - Physics Based Slide (2/2)*. [Video]. Available at: https://www.youtube.com/watch?v=TQsg8j-a3Y8&ab_channel=ReidsChannel [Accessed 21 January 2022].

thatgamecompany, 2012. *Journey*. [Computer game] West Hollywood, CA: Annapurna Interactive.

VR Playground. (2022). *Grappling Hook Virtual Reality Player in Unreal Engine*. [Video]. Available at: <https://www.youtube.com/watch?v=nyVDdEZXXsI> [Accessed 24 January 2022].