World Engine

Game Design Document

By BGH Projects

Contents

Executive Summary

World Engine is a virtual reality game intended to rehabilitate hand and arm burns patients. The game was conceived and built as a proof of concept solution to the lack of engagement of hand and arm burns patients in their prescribed rehabilitation programs. The mechanics of the game incorporate various movements that are required by the patient's prescribed rehabilitation programs, and the outer space theme and setting was chosen as a visually engaging environment in which the patient could participate in.

The game was originally intended to be played with a custom 3D-printed controller, however the game was adapted to be playable with a custom-built Arduino controller, which emulated the prescribed movements, as well as the standard HTC Vive controllers. The unique aspects of virtual reality that distinct it from traditional 2D or 3D games were utilised to create a unique UI/UX environment in order to facilitate an increase of participation from the patients. Various modes of gameplay were implemented, incorporating various forms of motivation including time-based pressure and creative exploration, to further the engagement from the target audience.

Project Brief

The project was initiated by my supervisor as a proof of concept solution to a problem that was present by an occupational therapist who was a colleague of my supervisor. The occupational therapist was working with hand and arm burns patients, and was prescribing a rehabilitation program to the patients that was required to avoid the skin and muscles in the affected hands and arms regrowing incorrectly. The incorrect regrowing of these body parts could result in a loss of ability to move and use said body parts. These rehabilitation programs involved a series of mundane and repetitive movements, including manually screwing and unscrewing a foot-long bolt by twisting their wrist, over a series of repetitions, as well as other gripping and squeezing motions. The occupational therapist was finding that their patients were neglecting to follow through on the program because they would lose interest through sheer bordem.

The solution that my supervisor proposed was to recontextualise the rehabilitation program through a gamified virtual reality experience. The movements required by those programs would be incorporated into gameplay, to ensure that the patients were properly rehabilitating. The project was designed for a team of two, a hardware engineer and a software engineer. The hardware engineer would design and construct a custom 3D-printed controller that would emulate the movements required by the program and enable these movements to transfer information to a software application. The software engineer would design and develop a virtual reality application that would be able to receive inputs from said controller and convert those to actions within the context of the game. The document concerns the latter part of that solution.

Ideation

Part of the virtual reality-based aspect of the solution was ideating a context/scenario in which the game would be played. I originally ideated 16 different contexts/scenarios in which the game would take place, including but not limited to roller-coaster design and testing, maintenance and repair of different sci-fi style robots, and the solving of custom-made 3D puzzles. The idea that my supervisor ultimately decided upon was that in which the player takes the role/ form of a celestial being in outer space, and must create habitable worlds for human beings to live on.

The rationale for this idea that was positioning the player in outer space would be a visually stimulating environment, and ideally an environment in which the player would want to enter and return to continuously. The concept of taking the form of a celestial being allowed the player to contextually enter an outer space environment without being locked into a space station or trapped in a space suit of some sort, viewing the environment in its entirety. The role of creating habitable worlds was conceived as a way of motivating the player to do something they probably haven't done before, and to give them a goal worth reaching, as will be detailed in later sections of this document.

UI/UX Design

The most exciting challenge about this project was the implementation of the user interaction and user experience requirements. This was the first project in which I had to create and implement a UI and UX for a virtual reality project, and I was excited to dive right into this novel problem space and utilise this unfamiliar technology. The key implementations of VR-related UI/UX design were the incorporation of the rehabilitation movements into gameplay, and the use of the VR space for the menus used in the game.

Movements

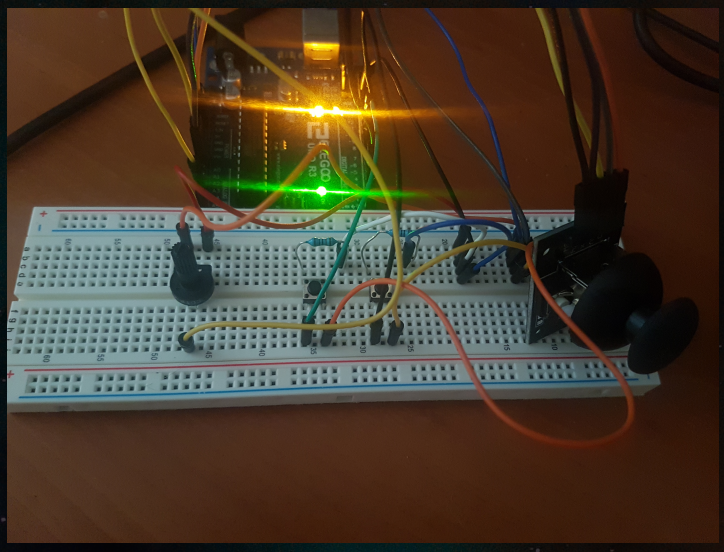
As discussed in the Project Brief, the requirements of the solution were to incorporate the rehabilitation movements prescribed by the occupational therapist into gameplay. There were three key movements that were required to be incorporated, which were:

- Twisting
- Gripping
- Pressing

Within the context of the game, these actions were required at certain points within gameplay to assist in the terraforming or planet forming process (depending on the game mode). Rather than have these movements directly correspond to tangible aspects of the planet forming process (e.g. twist this value to release a gas from a machine etc.), what the movements correlated to was abstracted away contextually. For example, the game might prompt the player to add water to a planet by simply twisting the controllers, without any direct contextual correlation between the movement and gameplay. This was to minimise the amount of 3D assets required for the game.

In regards to how these movements were to be implemented physically, as discussed in the Project Brief a custom 3D-printed controller was to be designed and developed which would contain different components which when interacted with would simulate the three rehabilitation movements. However, this controller was unable to actualised within the time budget prescribed for this project.

During the early development stage, in order to prepare for the integration of the custom controller's input into the game, I constructed a prototype controller out of an Arduino UNO R3 microprocessor, as that was the microprocessor that the custom controller would use. The prototype controller is shown below:



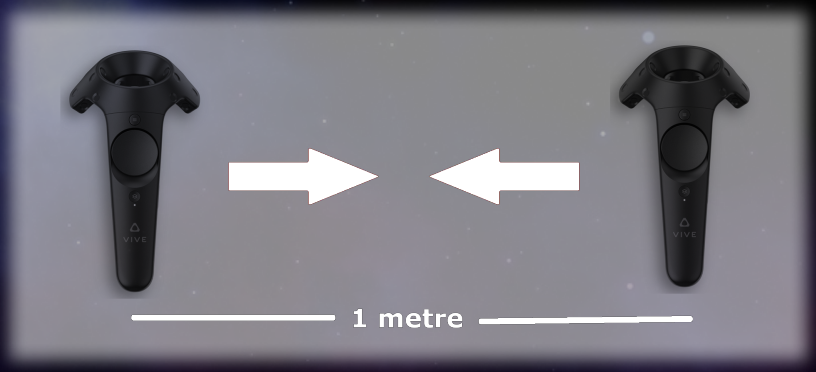
Prototype Controller

The secondary purpose of the prototype controller was to simulate the different rehabilitation movements, as feasibly as possible, using existing Arduino components. The potentiometer (left-hand side) was intended to simulate the twisting motion, the push buttons (center, below the green light) were substitutes for gripping motions, and the controller joystick (far right) was meant to emulate pressing. These components were primarily intended to test how the inputs were to be handled by the game, as their similarity to the actual movements was limited. Once this handling process was confidently integrated, the Vive controllers were used instead, in the absence of the custom controller.

Vive Controllers

When it became evident that the custom controller would not be completed on time, the Vive Controllers became the primary method of interacting with the game. Emulating the movements via the controllers presented different challenges.

The gripping motion was the easiest to implement, as the Vive Controllers come with grip buttons located near the palm of the user's hand. The pressing motion was implemented by requiring the player to spread the controllers 1 metre apart then bring them together within 20cm of each other. This constituted one press. For twisting, the rotation of the controller in 3D space was tracked, and one twist constituted the player twisting the controller until their palm was facing upwards and then twisting the controller again until their palm was facing downwards. These movements are communicated to the player in-game through the following images:

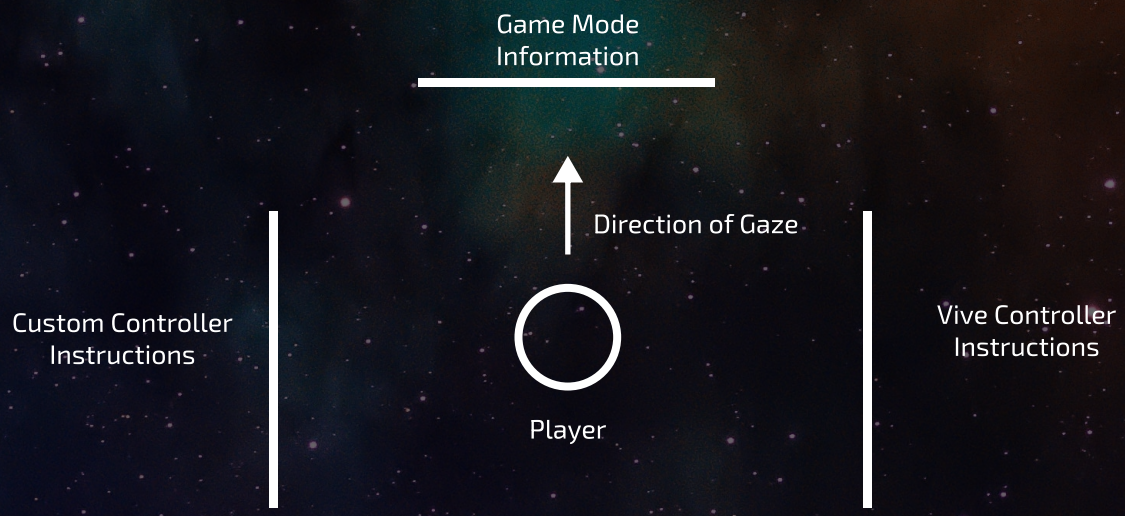


Pressing Motion Image

Twist Images

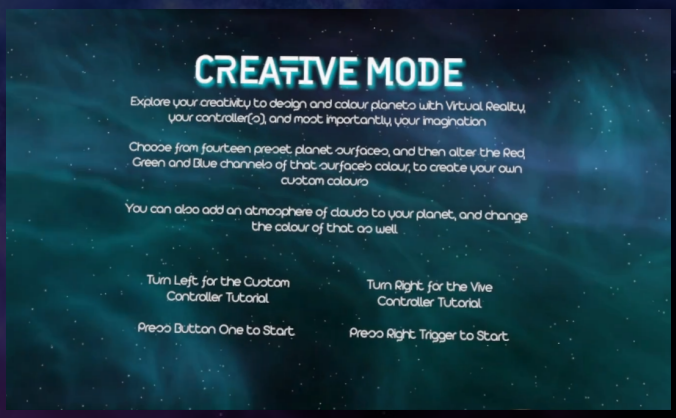
Menus

The menus that are experienced within the game were implemented to maximise the use of the three dimensional virtual reality space and the immersion that it offers. Within the pre-game menus, instead of just displaying all the information in front of the user to read, the information was divided into three sections; game mode rules, Vive Controller controls, and custom controller controls. These sections were displayed in front of, to the left, and to the right of the user respectively. This is visualised diagrammatically below:



Menu Implementation

The rationale behind dividing the information into three sections and surrounding the player with panels displaying the information was to immerse the player within the virtual world and enable them to look and move around the environment, contributing to their sense of engagement. Below are some sample in-game representations of these menus:



Creative Mode Menu
Front Display

Creative Mode Menu
Left Display

(NOTE: Since the custom controller was not implemented at the time, this side of the display was just placeholder)

Creative Mode Menu
Right Display

Game Modes

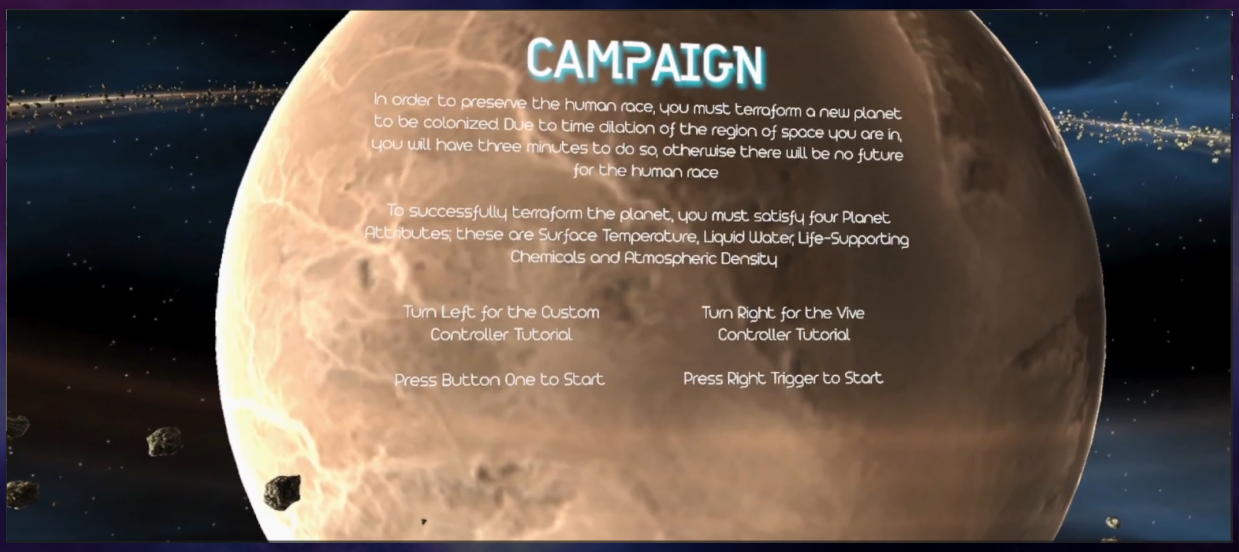
Four games modes were developed for World Engine. Each game mode provides a distinct method of engagement for the player, in order to engage them in multiple ways and to cater for a broader range of players. The game modes implemented are as follows:

Campaign

This mode provides a contextual method of engagement for the player. Before they begin playing, they are presented with a fictional scenario to provide them context to their gameplay. They are told that they have arrived at a potentially habitable planet which they must terraform in order to save humanity from extinction. They are also told that due to time dilation, they must terraform the planet within three minutes, otherwise humanity will be extinct. They are told that in order to terraform the planet, they must satisfy four Planet Attributes, which are:

- Surface Temperature
- Liquid Water
- Life-Supporting Chemicals
- Atmospheric Density

The player must perform one rehabilitation movement per Planet Attribute until that Attribute is satisfied in order to terraform the planet. See the next page for in-game screenshots of Campaign mode.



Campaign Mode Menu
Front Display

Campaign Mode Menu
Vive Controls

Campaign Mode Gameplay

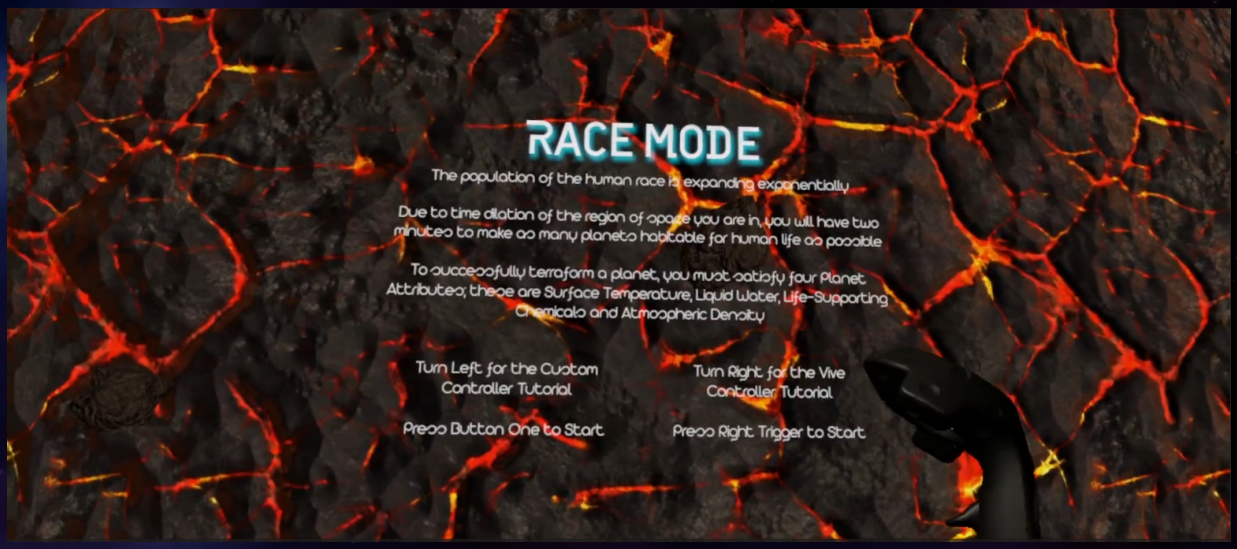
Increasing the Liquid Water Attribute by twisting the Left Controller

Campaign Mode Win

Race

This mode engages the user through the use of time constraints and the implementation of a scoring system. Players have one minute to terraform as many planets as they can. Each planet that they successfully terraform scores them one point. The intent behind the time constraint was to encourage a “rushing” mentality and to enhance gameplay through exhilaration as time runs down. The introduction of a scoring system gives the player a goal to aim for, and a goal to attempt to beat by replaying the game.

The core mechanics of the gameplay are the same as for Campaign mode, in that the player must perform one of the rehabilitation movements until a Planet Attribute is satisfied. Once all four Planet Attributes are satisfied for the planet, the planet disappears and a new one takes its place. Unlike Campaign mode, each Planet Attribute is randomly assigned a rehabilitation movement that must be performed to satisfy it, adding further variety to gameplay. See below for in-game screenshots of Race mode.



Race Mode Menu

Race Mode Gameplay

Race Mode Game Over Screen

Creative

This mode engages the user via a method that is alternative to objective-based constraints. Creative mode allows players to design, colour, and create their own planets. The mode intends to engage players through an avenue of exploration, enabling them to pursue their creative interests. There is no time limit or directed goal, allowing players the freedom to work towards their own outcomes.

In order to create the planet, the player is able to cycle through different preset surfaces, including planets of our solar system and other alien planets, colour the surface of the planet, add a cloudy atmosphere, and colour the atmosphere itself. This is enabled through the use of the different rehabilitation movements. Twisting the controllers allows the player to cycle through the surfaces. Pressing the controllers enables the atmosphere. Using the grip buttons, and the other buttons on the controller, is how the player changes the colours of different aspects of the planet. See below for in-game screenshots of Creative Mode:

Switching Surfaces by twisting the controller

Pressing controllers to enable
Clouds on the planet

Gameplay with all features enabled

Creative Race

This mode combines aspects of Race Mode and Creative Mode, to produce a unique experience in and of itself. Players are presented with a customised planet, similar to what can be produced in Creative Mode, along with a canvas planet, and they are required to colour and design the canvas planet to match the customised planet. Matching the planet involves identifying the same surface, surface colour, whether or not the planet has an atmosphere, and the colour of that atmosphere. Once the planet has been matched, the player scores one point, and a new customised planet is randomly generated. The player must match as many planets as they can before time runs out.

Players colour and design the planet using the same rehabilitation movements that are used in Creative Mode. Along with the creative aspect of Creative Mode, and the rushing mentality of Race Mode, this mode seeks to test the user's attention to detail, providing an alternative form of engagement. See below for different in-game screenshots of Creative Race mode.

Creative Race Mode Menu

Creative Race Mode Gameplay

Indication that the player has successfully matched the surface colour