Exploring The Future With Virtual Reality

UNREAL ENGINE

VIVE

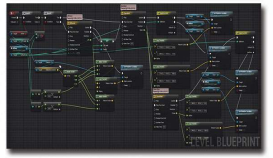
htc | STEAM VR

Client Info

Name: Mollie Smith.
Occupation: Professional Writer.
Interest: Technology, Karate.
Outcome for project: To give a good indication whether or not Virtual Reality is a viable option for developing feasible applications that contain storytelling elements.

Purpose

Virtual Reality technology provides us in the real world a gateway into the digital world. It gives its users the opportunity to do things, that if not were possible to do in the real world, would be possible through the digital world.
This project was undertaken with the purpose of researching the possibilities and the feasibility for the development of Virtual Reality applications, and also its possible use as a platform for storytelling.

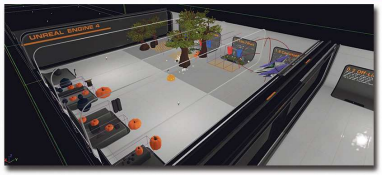


Unreal Engine

The Unreal Engine platform fit in perfectly for our needs especially with our prior experience and knowledge using this tool. Fortunately, Unreal had two major updates during our development cycle that added improved support and features for VR; further speeding up our development. Unreal has proven to be a viable product for developing VR applications. Many components of development can be done by non-programmers as the engine offers high-level methods to implement previously complex systems. Unreal offers two methods for programming: Blueprints and C++. Our projects were mostly developed using Blueprints which are a form of visual scripting, however, the programming principles remain the same.

Showcase

The showcase project provides a simple area free of distractions to demonstrate various aspects of the VR research and experimentation completed during this project. The level is designed in a simple layout where each element to be demonstrated is positioned on its own unique podium.



Tech Demos

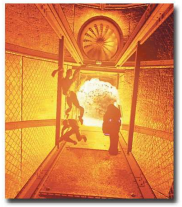
• This Archery Gallery Game allows the interaction of using two hands simultaneously to load, draw, aim, and shoot a bow and arrow at destructible targets. The arrow inherits bullet-drop physics to give it a realistic effect and allow the player to apply their own skill level to successfully hit a target, as one would in the real-world.

• The Basketball Game allows for a player to grab a ball and throw it in a hoop to achieve a score, the further the hoop, the higher the score. The ball has the Unreal Engine physics simulation applied to it that allows the ball to inherit life-like characteristic such as weight, bouncing, dropping.

• The Block Game is designed to give a user the freedom to teleport within a parameter and interact with any objects that are at reach. These objects can be picked up, thrown, and stacked on one another. It is entirely up to the user to decide how to interact with an object.

• The Breaking Game once again allows to pick up objects and throw them at targets that will shatter when within the range. I shows of the physics of the unreal engine and how players can turn an object into a weapon to cause a destructive impact on another object.

The Shooting Gallery is similar to the Archery Game but has given the user the ability to shoot a gun instead at any of the given targets. Interaction and skill levels change when using a gun like so, a gun shoots straight and its projectile reaches its target a lot faster than a bow and arrow.



Zombie Shooter

The Zombie shooter was designed to create a stress factor within the user to demonstrate how VR can play a role in triggering human emotions. The shooter consists of waves of Zombie hordes that can only be destroyed with a number of weapons within the players arsenal. The weapons include a pistol, an assault rifle, a shotgun, and lastly a grenade launcher. Each weapon behaves and deals damage in their own unique way. The player can choose which weapon they wish to help deal with enemy hordes, so long as the zombies do not get too close to the player where they would strike and damage the players health or potentially kill the player resulting in game over.

User Interface

- User Interfaces should feel like an integrated part of the 3D world the user is in.
- Do not require the user to swivel their eyes around to view the UI, this can cause eye strain and simulation sickness, try to position the UI in such a way that it sits in the user's field of view.
- User caution with elements of the UI that scale with the position of the user's head (Things that get bigger or smaller as the head moves further or closer to the object) If done wrong, this can cause simulation sickness. If it is to be implemented ensure that it responds accurately to the user's head movements.

Controlling the Character

- As user input cannot be viewed when wearing a headset, allow the use of familiar controllers and controls as the default method of input. Remember that if a keyboard is required, the users will have to rely on tactile feedback or try keys to find the controls.
- Maybe consider using head movement as a method of control.

Sound

- When using audio, keep in mind that the source of the sound will follow the head movements when they wear headphones, but not when the user uses speakers. Allow the user to choose their output device whilst in game, and ensure that sounds come from the correct locations with the headset relative to the output device.
- Ensure that positional tracking is working when designing sound. Sound should become louder as the user moves closer to the source.

Rendering

- Using the default distortion shaders is a must. Using estimates and approximations, even if it looks or feels about right, can prove to be discomforting for the user.
- Maintaining complete immersion from the beginning to end is a must. For example, putting a giant splash screen that won't react to head movements is a bad idea
- Think about looking at super sampling and or anti-aliasing (a method used to remove jagged and pixelated edges on rendered objects) to fix apparently low resolutions, which will appear to be worse closer to the center of the eyes.

Minimizing Latency

Latency refers to the "jumpiness" of a video game, if the image that appears on screen appears to be choppy or distorted it can cause problems for the user

- Ensure the games framerate runs at or higher than the refresh rate of the headset (in the case of the Vive, 90fps) It is a good idea to ensure a buffer in fps in case there are any framerate drops that occur as the game is played

Optimization

- Do not drop the display resolution, although it seems like a cheap and easy way to improve the games performance, lower resolutions can lead to extreme discomfort for the user including simulation sickness and increased eye strain.
- Decreasing the eye render buffer resolution is a good way to save on video card memory and increase the framerate of the game

Positional Tracking

- Try not to use any visuals or effects that will affect the user's grounding in the environment. What is meant by this is try not to make them feel off balance. Moving the horizon line in the user's field of view or other large components will conflict with their real world sense of self-motion will lead to extreme discomfort and can be highly off-putting for the user.
- The display should always respond to the user's head movement, no matter what, including of the game is paused or if they are simply browsing menus. The user should always be able to look around.

Movement Speed

- Not moving in VR is the most comfortable experience for the user, however, if the user is required to move through the environment, it is recommended that the user moves at a constant velocity.
- Teleporting between locations is a really good idea for movement as the player remains stationary and the world won't have the effect of being moved around them.
- Moving in one direction whilst looking in another direction can be disorienting for the user. Try to minimize the need for the user to look away whilst looking in another direction especially if the user is moving faster than a walking pace.

Cameras

- Zooming the camera in or out can induce or worsen simulation sickness. Do not use zoom effects.
- For any third person content all guidelines for acceleration and movement still apply, just because the user is not in the character's body so to speak, does not mean that these do not apply.
- Do not use any head bobbing effects, ever. This is because they are small accelerations in different directions as the character is moving.

Simulation Sickness

Simulation sickness is a form of induced motion sickness which can occur whilst playing any virtual reality game.

- Virtual reality developers are the worst testers as they can become acclimatized to the effects of simulation sickness.
- It is advised to test all content in a variety of ways, especially with unbiased users in order to assure a comfortable experience for a wide range of users.
- People will respond to and experience simulation sickness differently, they may never go through the experience or everything could be literally unplayable for them. They may respond to visually induced motion sickness with the headset on more than they do with a regular computer screen. Your users should not be expected to power through any intense experience, nor should they be expected to.
- Consider implementing ways for your user to change the visual experience, adjusting the intensity of things. For example, movement speed, acceleration size, and the width of the displayed field of view. If you can alter these settings, they should default to the lowest

Micheal Dunstan - Max Sheahan - Dylan Knapp - Kieran Bennet - Kane Maxwell