Spooky McSpook: A Virtual Reality RPG for Improved Situational Awareness

CHI MINDY ROBIN, DĄBROWSKI JAN KRZYSZTOF, KENNEDAL AXEL,
LUUSUA EMIL JAN OTTO, SIM LONG SIANG, VU DUC LONG, YAO KAN

INTRODUCTION

Situational awareness skills can make a difference in everyday life, whether in work or just going about our days. Our project helps users improve these skills with an immersive, stimulating VR RPG game experience.

Socio-economic forces are increasingly pushing us towards optimizing the relationship between effort and resulting value. In general, the global industry is shifting more from physical to mental work. We also see that as our society becomes safer, develops more technology, and relies more on mental work, we lose our physical awareness. Often we find ourselves walking and waiting with eyes glued to our smartphones. Even at the gym, we tune into our music and tune out of the world around us.

Spooky McSpook is a VR RPG open world game with the element of exploration. The main gameplay involves encountering enemies in the world, and discovering new weapons to fight with. To avoid being ambushed by mad skeletons or other evil creatures of the night, the player must always be aware of their surroundings. Constantly having to look around in the virtual world will enforce a habit that transfers to real-life and helps the player realize the importance of situational awareness.

Methodology - Tools

We use Unity and C# scripting. Our hardware used during development is two desktop PC setups in the VR room at The Hive on the NTU campus. One of these desktop PCs is equipped with an Oculus Rift headset and controllers. The other desktop has a HTC Vive including controllers. Steam VR is our interface to use these VR systems with Unity, providing handy prefabs such as the [CameraRig] which makes the game camera follow the movement of the player's headset and also includes controllers. Additionally, the VRTK toolkit gives us some extra VR functionality such as picking up objects and using these. Some development work has also been carried out on our laptops, using the VR simulator by utilizing the VRTK_SDKManager seamlessly switching which SDK is used depending on the setup used. Our 3D assets and animations are primarily sourced from the Unity Asset Store.

Methodology – Design and Development

The player is free to move around in a large, dark forest and can equip a simple melee weapon at the start of the game. Weapons are controlled with arm movements - if the player is wielding a sword he may wave it around as if he was holding it in real-life. Landmarks with better weapons can be found around the map. There is also a magic staff that will allow the player to fire projectiles with a trigger on the controller, these are also aimed using the orientation of the controller.

A few areas of the map are dangerous. Enemies will spawn when the player wanders too close to tombs which are scattered throughout the arena, as well as some enemies being present from the start of the game. The player will encounter many different kinds of enemies which all have unique behaviors and game mechanics which makes the game experience diverse and interesting. The enemies also has different amounts of health, but are all damaged the same way if they are hit by the player's weapon or projectiles.

At first the goal of the game is to defeat a number of regular enemies. Once enough enemies have been killed a final boss will be summoned. The player has a limited amount of health, and must defeat this boss before they take too much damage from enemies to achieve the epic victory royale.

The availability of both melee weapons and a ranged weapon adds a strategic element to the game. For example, the player can choose to opt for the powerful ranged weapon, but might lose a lot of health in the process of retrieving it since it is well guarded and might be in trouble if enemies manage to get close.

IMPLEMENTATION AND EXPERIMENTATION

Movement

Player movement is not by teleportation but using the control stick for free movement. Although this has a risk of inducing motion sickness, it allows for a much more fluid and immersive gameplay experience. This is especially important in an RPG environment involving close range combat where being able to run away from a horde of enemies while simultaneously fending of the closest ones is a regular occurrence. Using traditional ways of moving around in VR such as teleportation or swinging your arms back and forth would not suffice in this context since you need to be able to swing around the controllers to fight while simultaneously moving around independently of these motions.

Weapons

Weapons are picked up by reaching for them with the VR controller and pressing the grab button. This will attach them to the controller in a preset orientation such that they match how the player would wield them in their real-life hands and swinging them around feels intuitive. The magic staff fires projectiles by pressing the trigger button on the controller holding the staff, these are fired in the direction the staff is currently aimed. The staff and projectiles also have light sources integrated into them, which makes them useful to light up the dark environment while also giving them a magical feel.

Enemies

Enemies are animated with their behaviors through Unity's Animator to easily tie them with script variables. This allows smooth transitions between animations as different events occurs, such as their health reaching zero resulting in a death animation regardless of what the enemy was doing previously. The following enemy types have been implemented:

1) Skeletons. These will blindly chase the player if they get close enough, and otherwise idle in a stationary manner. They damage the enemy by swinging their sword, which takes some time to do enabling the player to dodge the hit if fast enough.

2) Zombies. These are scripted to act as 'hordes' rather than independent entities, they wander around aimlessly but stay as a group. If one gets close to the player, they will not only start chasing, but also alert all nearby zombies to do the same. To avoid the player being completely overwhelmed by these hordes, the zombies have been given a low amount of health so that each one can quickly be killed.

3) Ghost wizards. These are more powerful enemies that haunt certain areas of the map, being bound to these areas they will teleport back if they manage to chase the player out of it. They have also been given special animations when damaged (blood splattering) and when hitting the player (fire explosion) to make it apparent they are a kind of "mini-boss" and should be approached with caution.

4) Poisonous mushrooms. These are stationary enemies rooted in a single place, but has a ranged attack that will poison the player if hit. The poison has a damage over time effect and clouds the player's vision for a short period of time.

5) Dynamite bunnies. These bunnies are not hostile beings, and will peacefully jump around the map not looking for any trouble. Though due to the dynamite strapped to their back, if the player happens to step on one it will explode and deal a large amount of damage.

6) The boss. A very large golem, which has three different attacks it will randomly pick among. It also has a large amount of health, which makes it a real challenge to deal with. As the boss spawns, dramatic battle music is played to instill the feeling of dread into the player.

The different enemy behaviors are controlled using scripts which is also responsible for moving them around the map. Scripts also control the spawning of enemies, and are easily customizable to spawn enemies in a certain range, with a specific enemy type, and with a certain frequency with some randomness.

Combat & Health

One of the major challenges unique to our project was to make the combat feel natural and fair. This was accomplished by having damage be dealt only as specific body parts of the enemies collide with the player, such as the sword of the skeletons or the hands of the zombies. Special colliders had to be created for this purpose, the aim of these was not to necessarily match the shape of the enemy but instead make it is so that if the player perceives they are hit, the collider will have hit them. This results in the colliders being bigger than the mesh of the enemies' damaging body parts, since they seem to be further away than what is often perceived by the player and thus actually "missing" the player hitbox.

To have more clean and modular code, our health and damage system are managed by scripts that can be attached to any Game Object. It's easy to define how much health a unit has, and also define which types of Game Objects should cause damage to the unit to avoid things like friendly fire. This is implemented by use of tags attached to Game Objects which should deal damage, such as "PlayerWeapon" and "EnemyWeapon". Using such a script also makes the amount of damage to deal during gameplay a variable, this is utilized when an enemy dies by setting their damage to zero. This avoids situations where for example the player steps on the corpse of a defeated enemy and gets damaged because they collided with the enemies weapon.

Design of User Interface

When launching the game the player is greeted by the pause menu, which contains the logo/name of the game and instructions for the controls.

This UI follows the player's head movements as to be easily accessible.

A basic HUD also follows the player as they wander around. This displays the remaining health of the player, as well as how many basic enemies have been killed.

Enemies also have health indicators floating above them. For picking up weapons, the controls will be outlined.

CONCLUSION

In an easily accessible and beloved format of an RPG, our game provides a fun and familiar way of practicing situational awareness in a virtual environment. With the use of VR and many technical innovations such as motion-controlled weapon combat and unique enemy behaviors we are brought closer to an intuitive and immersive method of practicing real life skills. The 360 view of the world requires good spatial management for the player. There is high potential for using the technology that brought us away from the real world to bring us closer back to it.