

Julien : Musique dynamique

Raphaël : Génération procédurale

Pascal: Personnalisation des avatars

## Audio

Le contenu sonore du travail est répartie parmi les fichiers selon les types d'usage des sons. La musique et l'ambiance sonore se trouve dans des classes réservées pour la musique alors que les effets sonores et les jiggles sont dans les classes de gestion pour les situations qui les utilisent. Il y a un paramètre de volume pour chaque type de son (musique, sfx et ambiance). Ceux-ci sont enregistré dans le `playerRefs`, permettant de les conserver entre chaque session.

L'ambiance sonore consiste d'une trame d'un courant d'air. L'environnement étant un bâtiment abandonné par les vivants, cela donne un aspect de désolation.

Les différents effets sonores consistent de sons de tir lorsque le joueur tire, ce qui inclut la collision de la balle à un mur et de la douille qui tombe au sol. Des cris ont été ajoutés aux zombies lors de leurs attaques.

Des pistes distinctes sont utilisées pour le menu principal, le menu de pause/conception de personnage et le gameplay. Il y a aussi un changement de musique lorsque le joueur se retrouve avec peu de vie ou lors de sa mort. Il y a également un jiggle pour chaque 5 zombies abattus qui monte en tonalité. Lors du gameplay, un crossfade est utilisé entre les musiques de jeu et du menu de pause avec l'aide de coroutines.

## Génération procédurale

L'environnement est généré procéduralement au lancement de la partie. Celui-ci consiste en un labyrinthe sans aucune boucle et sans sortie. La structure est générée à partir d'une grille avec la méthode par fusion de chemins. Chaque case de la grille reçoit un identifiant. L'algorithme en choisit un au hasard parmi une liste, cherche parmi les cases avoisinantes une autre ayant un identifiant différent et si elle existe, crée un chemin entre les deux et assigne l'identifiant de l'un à toutes les cases ayant le même identifiant que l'autre. Sinon, la case est retirée de la liste. L'algorithme se poursuit jusqu'à ce qu'il ne reste qu'un seul identifiant pour toutes les cases. Cette méthode garantit qu'il existe un chemin allant de n'importe quelle case à une autre.

Une fois les chemins générés, toutes les cases possèdent une liste d'adjacence. À l'aide de cette liste, un mur est générée entre chaque cases avoisinantes absentes de la liste. Des coins sont ajoutés pour ajouter de l'épaisseur aux murs. Les murs et les coins sont générés grâce à l'instanciation d'un modèle de départ.

Les murs utilisent une texture simple qui se combine bien bout-à-bout. Les coins utilisent la même texture. Celle-ci est écrasée, mais donne un effet différent désiré. Un modèle légèrement différent est parfois généré, donnant un peu plus de relief à l'environnement.

Les monstres quand à eux sont générés de façon simplement aléatoire à une distance précise du joueur. Pour optimiser leur utilisation, lorsque ceux-ci se retrouvent trop éloignés du joueur, ils sont téléportés à une certaine distance de celui-ci. La distance est toujours plus grande que le maximum que peut voir le joueur, de sorte qu'il a l'impression que les ennemis sont partout, tout en gérant efficacement les ressources.

Finalement, l'environnement de départ est 100 % reproductible. Avant de débiter une partie, un champ permet de fournir un germe servant à la génération de cet environnement. Le même germe donnera toujours les mêmes conditions de départ, soit la position des murs, du joueur et des premiers ennemis. Une fois la séquence principale amorcée, la génération des ennemis dépendra de la position du joueur.

## Personnalisation des avatars

L'avatar du joueur possède plusieurs options de configurations. D'abord, il possède plusieurs modèles préconstruits, dont certains avec des accessoires. Ensuite, on peut modifier la grandeur et la largeur du modèle. La grandeur modifie de la même façon tous les composants de celui-ci, tandis que la largeur (ou body type) ne le modifie que dans une seule dimension, donnant une allure plus maigre ou au contraire plus large et plus robuste.

Après le modèle de base et la forme, les couleurs sont aussi modifiables. Chaque modèle possède deux composantes de couleur de vêtement qui sont appliquées et recombinaison sur le modèle final. Le reste du modèle est géré par une 3<sup>e</sup> option de couleur, celle de la peau. Elle applique un filtre qui modifie la teinte légèrement, tout en gardant les traits originaux, tel que les yeux, la barbe et les cheveux, s'il y a lieu.

Le résultat final est un masque appliqué sur le modèle sans problème.