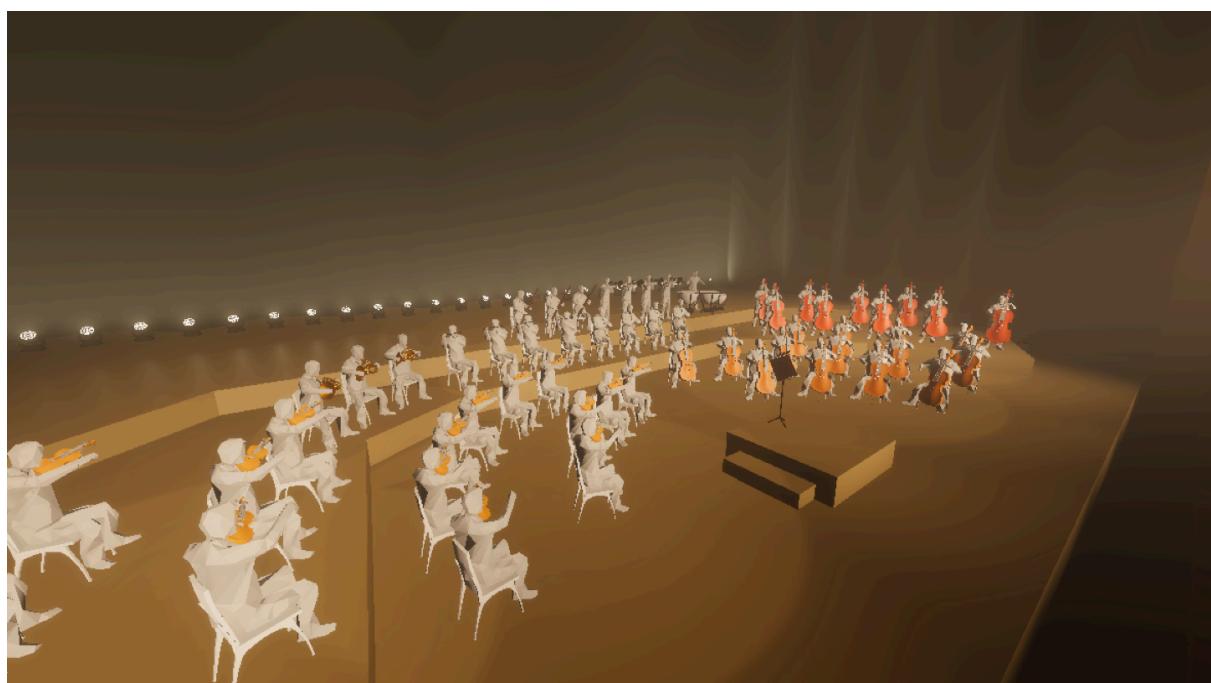


Rapport du projet “Chef d’orchestre”

Projet RV01 – A24



Céline Ozkaya et Adrien Hervé
09/01/2025

Sommaire

Sommaire	1
Introduction	2
Environnement 3D	3
La salle de concert	3
L'orchestre	3
Le chef d'orchestre	4
La salle d'attente	4
Interactions	5
Début du morceau	5
Sélection de l'instrument	5
Indication du tempo	6
Indication de l'intensité	7
Solo d'un groupe	7
Fin du morceau	7
Gestion et contrôle de la musique	8
FMOD	8
Fonctionnement	8
Immersion du joueur dans la scène virtuelle	9
Niveaux de réalisation	11

Introduction

Ce projet s'inscrit dans le cadre de l'UV RV01 à l'automne 2024. Il est réalisé par Céline Ozkaya (GI04) et Adrien Hervé (GI07) et suivi par Indira Thouvenin et Yohan Bouvet.

Nous proposons à l'utilisateur d'incarner un chef d'orchestre guidant les musiciens pendant l'interprétation de la Symphonie n°9 « Du Nouveau Monde » d'Antonín Dvořák, 3e mouvement. L'utilisateur dispose ainsi d'une grande liberté quant au choix des nuances et du rythme adoptés par les musiciens qu'il dirige. Pour ce faire, il dispose d'un casque et de contrôleurs HTC Vive.

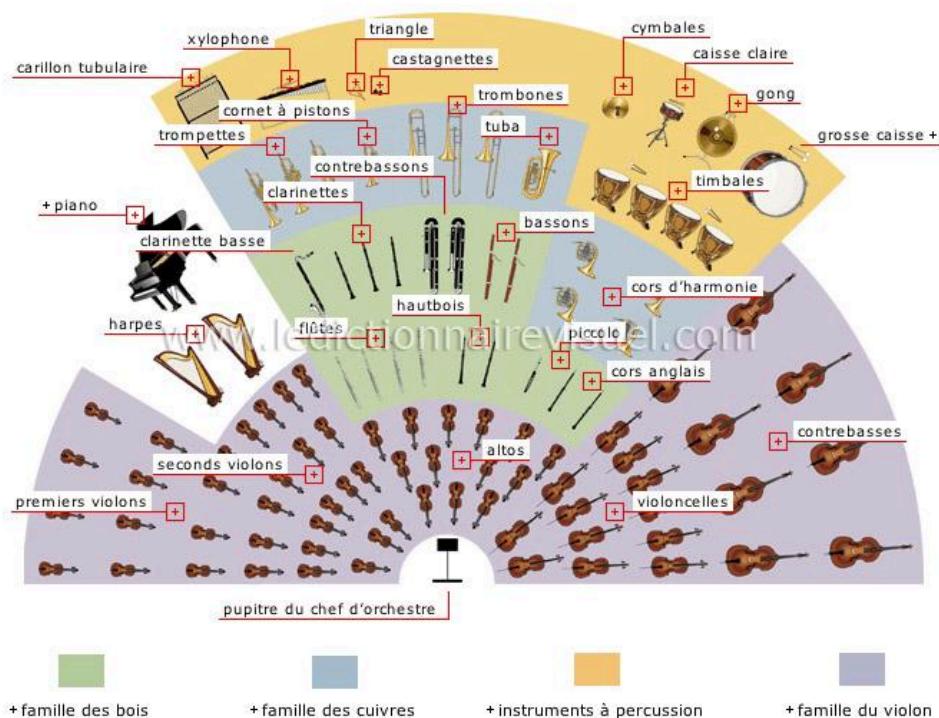
Environnement 3D

La salle de concert

L'environnement 3D a été réalisé sur Blender, Capture et Unity. Dans un premier temps, nous avons fait le modèle 3D de la salle de concert sur Blender. En parallèle, nous avons utilisé le logiciel Capture pour modéliser les structures d'accroche, ainsi qu'exporter de nombreux modèles 3D utiles pour notre projet (instruments, sièges, projecteurs au fond de scène). Une fois tous ces fichiers disponibles, nous avons tout assemblé sur Unity, où on a également ajouté les personnages, les sources lumineuses, le modèle du système de sonorisation, etc.

L'orchestre

Le jeu permet à l'utilisateur d'être immergé dans une salle de concert. Pour ce faire, l'orchestre est représenté face à lui et divisé en plusieurs sections représentant les instruments impliqués dans le morceau à diriger. Par exemple, les violons sont regroupés dans une première section, les hautbois dans une autre, etc. à l'image de la figure 1. Le joueur peut ainsi diriger tous les instruments d'un même type (ex : violons) en même temps.



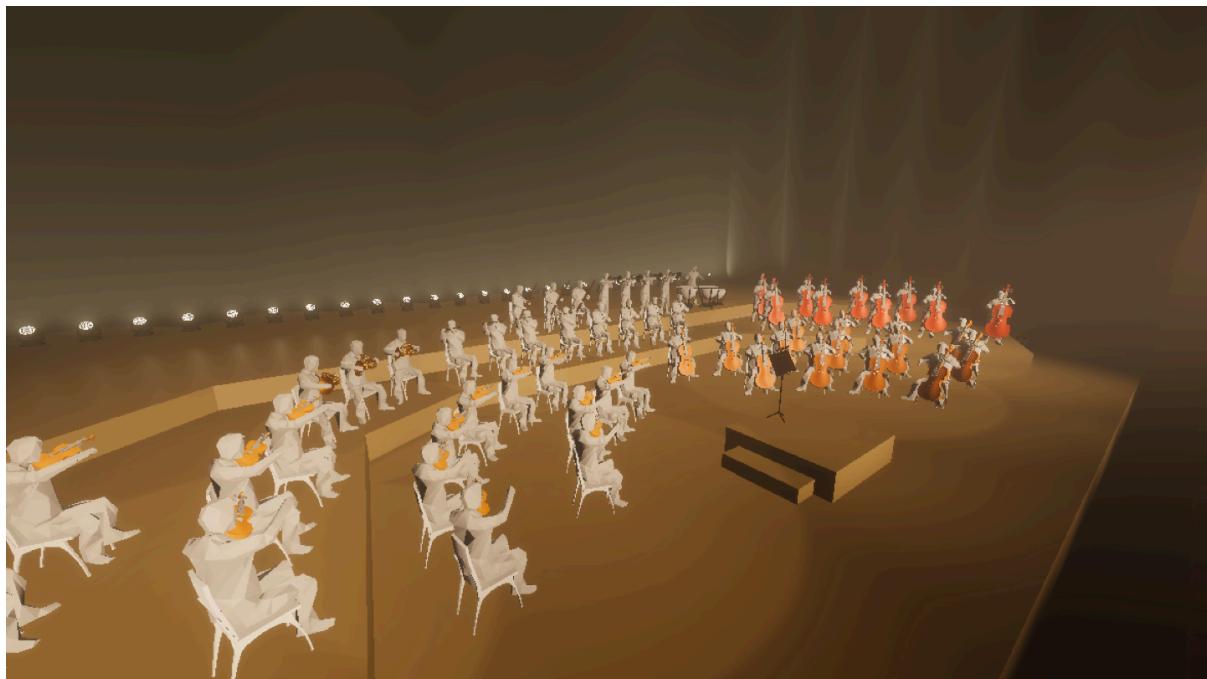


Figure 2 – Scène du projet

Les instruments représentés sur la scène (fig. 2) correspondent à ceux que l'on entend dans le morceau que le joueur dirige.

Le chef d'orchestre

Le chef d'orchestre est représenté par les contrôleurs qui matérialisent ses mains. Une baguette est également présente dans sa main droite.

La salle d'attente

Nous n'avons pas eu le temps d'implémenter de salle d'attente. Au commencement de la partie, le joueur est placé sur son estrade face à l'orchestre.

Malgré l'absence de la salle d'attente, nous avons fait en sorte que le jeu ne commence pas immédiatement afin de permettre à l'utilisateur de se familiariser avec l'environnement. Pour ce faire, au lancement du jeu, le joueur arrive sur la scène et peut entendre les musiciens accorder les instruments. Il peut alors se déplacer jusqu'à son pupitre ou bien explorer la scène sans se soucier du début du jeu. Ensuite, quand il souhaite commencer, il lui suffit de tapoter sa baguette sur le pupitre. Les musiciens cessent de s'accorder et le silence règne quelques secondes avant que le morceau ne démarre.

Interactions

Début du morceau

Le début du morceau est demandé par le joueur en tapotant sa baguette (contrôleur droit) sur son pupitre.

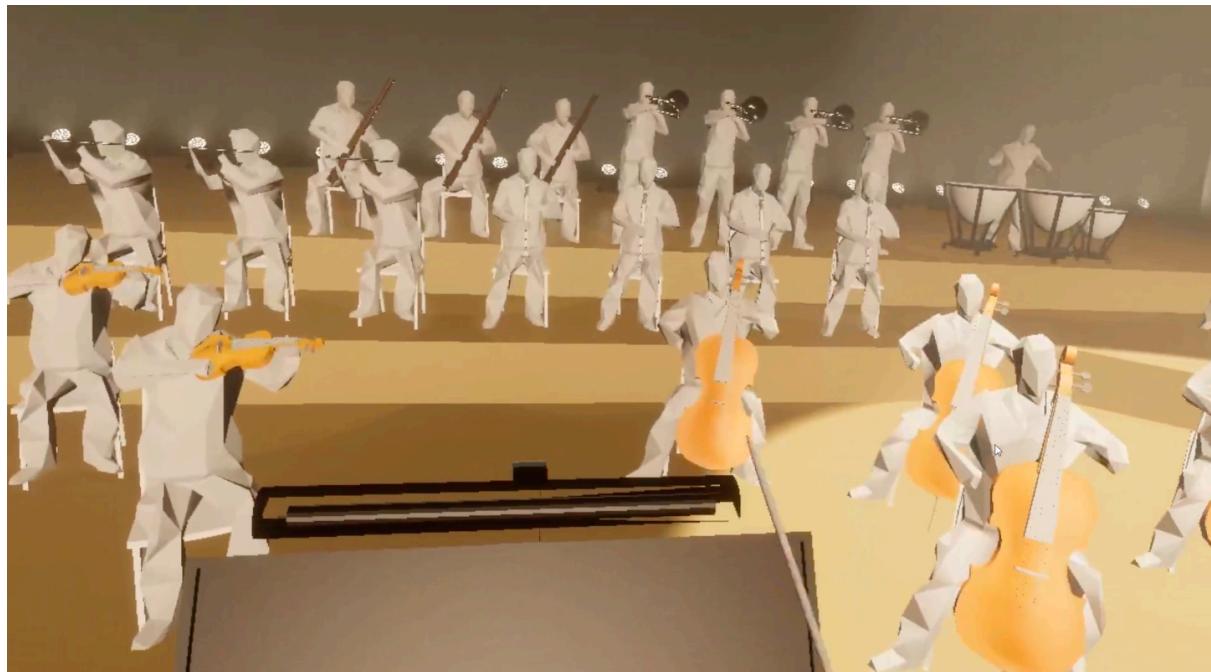


Figure 3 – Début du morceau

Sélection de l'instrument

Lorsque le joueur balaye du regard l'orchestre, chaque groupe d'instruments (ex : violons, flûtes, etc.), voit son spot lumineux associé s'allumer un à un afin d'indiquer explicitement lequel est observé. Quand le regard quitte le groupe d'instruments, le spot s'éteint. Pour ce faire, nous avons implémenté du Raycasting (script/Raycasting.cs/Raycasting()). La synchronisation entre les instruments et les spotlights a été faite avec des tags et des layers.

Quand un joueur veut sélectionner un groupe d'instruments, il doit poser son regard sur l'un deux, ce qui allume le spot, et cliquer sur le Grip droit. Le fait de cliquer, allume le spot jusqu'à ce que l'on clique à nouveau pour visualiser le groupe sélectionné. L'objet représentant la sélection est récupérable via son getter. Cela a été implémenté dans une fonction OnGrab(). On peut alors décider de l'intensité associée.



Figure 4 – Sélection du tambour

Indication du tempo

La modification du tempo est appliquée à tous les instruments pour assurer la cohérence du morceau. Cela est implémenté dans le fichier script/Tempo.cs.

Pour ce faire, l'utilisateur clique sur la gâchette (fonction OnTrigger()) ce qui active un objet Tempo (fig. 5) placé à bout de bras du joueur. Pour déterminer le tempo, l'utilisateur traverse l'objet avec le rayon de son contrôleur, chaque passage étant matérialisé par un retour haptique (vibration). Au premier passage à travers l'objet, un timer est lancé et le nombre de passage est compté. Lorsque la gâchette est à nouveau déclenchée, le tempo est calculé (nombre de passages / timer). Ainsi, plus le joueur traverse l'objet, plus le tempo est rapide. La valeur tempo est récupérable via un appel à son getter.

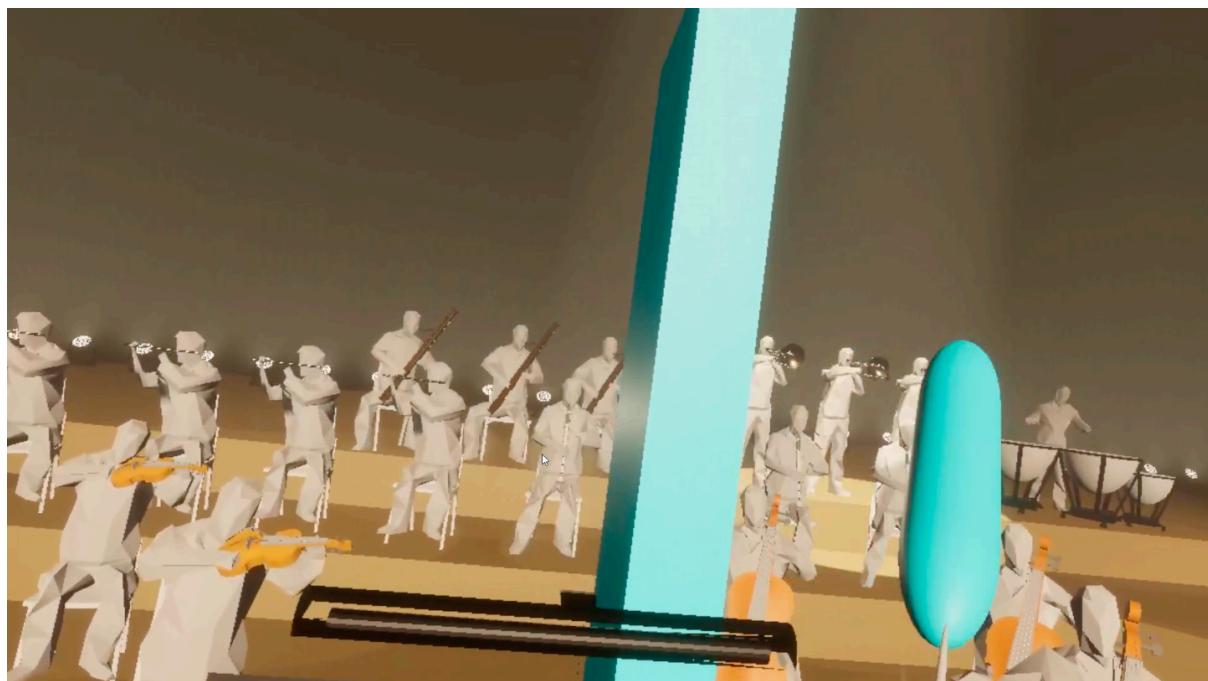


Figure 5 – Choix du tempo

Indication de l'intensité

Cela est implémenté dans le fichier script/Intensite.cs.

Après avoir sélectionné l'un des groupes d'instruments, le joueur peut indiquer l'intensité avec laquelle jouer via un Trigger sur son contrôleur gauche. Il lève le contrôleur pour augmenter l'intensité et baisse le contrôleur pour diminuer l'intensité.

Solo d'un groupe

Cela est implémenté dans le même fichier.

Après avoir sélectionné l'un des groupes d'instruments, le joueur peut choisir de couper le son de tous les autres en levant le bras au-dessus de la tête.

Fin du morceau

Pour indiquer la fin du morceau, l'utilisateur doit écarter les bras. Cela relance alors la séquence de départ avec l'accordage des musiciens, et le joueur peut recommencer l'expérience s'il le souhaite.

Gestion et contrôle de la musique

FMOD

FMOD est un logiciel de type DAW (Digital Audio Workstation) qui permet d'assembler des pistes audio et les modifier à l'aide de plug-ins de traitement de signal (réverbération, modification de la hauteur par exemple).

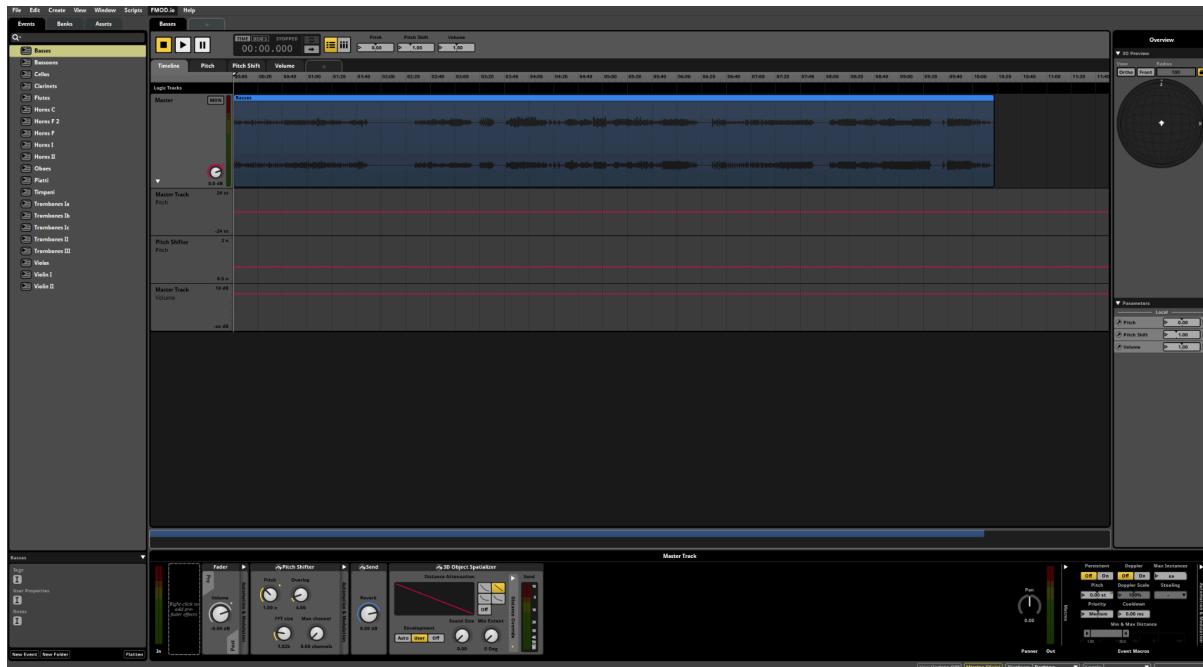


Figure 6 – Interface de FMOD Studio

Nous avons choisi ce logiciel car il offre une interface facilement utilisable sur Unity grâce au package FMOD.

Fonctionnement

Chaque piste d'instrument (créé au préalable grâce à des logiciels spécialisés tels que Ableton et Komplete Kontrol), est ajoutée à FMOD dans ce qui est appelé un Event. Dans cet Event, on ajoute des paramètres éditables qui modifient des propriétés audio de ce qui est joué : le volume, la vitesse et la hauteur du son (qui permettra de compenser la modification liée au changement de vitesse de la piste). Ces paramètres sont ensuite accessibles sur Unity et sont modifiés par les scripts de gestion des interactions. Les Events sont utilisables dans unity en instantiant un script EventEmitter, dont l'une des variables d'entrée est le nom de l'Event dans le projet FMOD.

Immersion du joueur dans la scène virtuelle

Nous avons cherché à rendre l'immersion du joueur la plus naturelle et réaliste possible.

Pour ce faire, une salle de concert grandement réaliste a été réalisée en jouant sur les lumières, la fumée et la profondeur pour rendre compte des sensations convoquées dans un tel contexte (fig. 7 et 8).

De plus, le morceau et ses pistes ont été spatialisés sur la scène afin que le joueur perçoive d'où le son provient. Ainsi, la piste du violon est réellement issue de la section des violons.

Par ailleurs, une fonctionnalité de retour haptique via la vibration d'un contrôleur a également été ajoutée afin de matérialiser le choix du tempo qui aurait été très abstraite sinon. Cela renforce l'ancrage du joueur dans la scène.

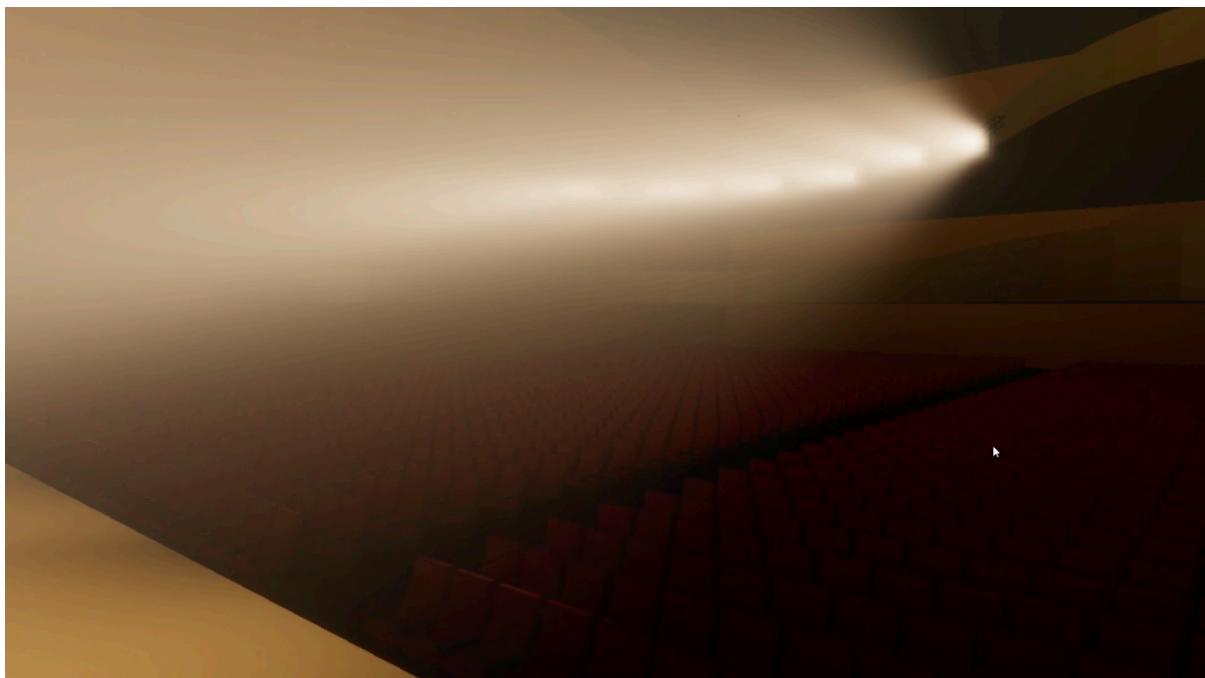


Figure 7 – Fond de la salle de concert



Figure 8 – Spot lumineux et hauts parleurs

Niveaux de réalisation

Nous avons réalisé une combinaison des deux premiers niveaux décrits dans le cahier des charges.

Certaines tâches, comme l'ajout d'un second morceau, ont été abandonnées car elles ne nous semblaient pas importantes au regard des attentes de RV01. En ce sens, nous avons choisi de travailler en priorité sur le développement d'interactions pertinentes (raycasting, collision, déplacement, trigger, etc.) ainsi que sur l'immersion du joueur dans la scène. Le tableau ci-dessous résume les différences entre le cahier des charges et les réalisations :

Objectifs	Cahier des charges	Réalisation	Interaction
Début du morceau	Oui (lever les bras)	Oui (repensée)	Taper la baguette sur le pupitre
Fin du morceau	Oui (cesser les mvt.)	Oui (repensée)	Ecarter les bras
Salle d'attente	Oui	Non	/
Selection des instruments	Oui (regard)	Oui (repensée)	Raycasting + Grip
Tempo du morceau	Oui	Oui	Trigger + collisions
Intensité par instrument	Oui	Oui	Trigger + Déplacement vertical du contrôleur
Solo d'un instrument	Non	Oui	Lever un bras
Environnement 3D	Oui	Oui	/
Retour haptique (Pour le choix du tempo)	Non	Oui	Vibration du contrôleur

Tableau 1 – Réalisations au regard du cahier des charges proposé