
Projet de session individuel ou en équipe de 2.

I. Description :

Le projet consiste à réaliser un jeu vidéo dans un environnement 3D rendu en temps réel à l'aide de la plateforme Unity. Le jeu se déroule à l'intérieur d'un labyrinthe (environnement intérieur) ou sur un terrain (environnement extérieur) ou encore dans un environnement hybride (intérieur et extérieur). Le joueur est représenté par un avatar animé (objet 3D, personnage, vaisseau, etc.) qui se promène dans l'environnement 3D afin de réussir sa mission. Il peut avancer, reculer, tourner à gauche ou à droite. Il sera impossible pour le joueur de passer à travers les murs ou les différents objets présents dans l'environnement.

L'environnement doit contenir, au minimum quatre objets interactifs qui peuvent avoir une action immédiate ou encore être ramassés par le joueur qui pourra les utiliser plus tard. Leur activation peut se faire au simple contact du joueur. Elle peut être aussi déclenchée par une touche du clavier ou un autre mécanisme de votre choix. Dans tous les cas, une rétroaction (visuelle ou auditive) est donnée à l'utilisateur lorsqu'il activera ou lorsqu'il ramassera un objet. Le choix de la nature des objets ainsi que leur effet dans l'environnement sont laissés à vous. En guise d'exemple, il peut s'agir :

- D'une boisson permettant d'augmenter la vitesse de déplacement de l'utilisateur pendant un certain temps,
- D'un levier permettant d'ouvrir une porte ou de faire descendre un mur, d'une clé pouvant être utilisée plus tard pour ouvrir une porte,
- Etc.

Le jeu doit intégrer l'utilisation d'un compteur qui sert à afficher des points, le nombre d'objets qui reste à ramasser (ou qui ont été ramassés), le niveau d'énergie du personnage, le temps écoulé ou restant à la partie, etc. Pour remplacer le compteur en chiffre, vous pouvez utiliser une représentation graphique (ex.: barre de vie, de vitesse, d'énergie, de temps, etc.)

Le jeu doit comprendre trois scènes :

1. Une scène introduction (Menu du jeu)

Elle est présentée au joueur au début du jeu sous forme de menu. L'interface est générée à l'aide des éléments d'interface utilisateur (UI). Il faut y placer, entre autres, des boutons permettant au joueur de commencer une partie et de consulter le tableau des meilleurs pointages (si présent), un champ de texte permettant de saisir le nom du joueur. Le nom sera affiché dans la scène de conclusion. Il faut aussi mettre en bas à droite du menu, votre nom et la date de réalisation pour votre droit d'auteur.

2. Une scène jeu pour le déroulement de la partie.

3. Une scène conclusion

Elle sera affichée dans l'une des conditions suivantes :

- Le temps du jeu est écoulé
- La mission est réussie
- Ou autre chose (de votre choix)

II. Autres éléments à inclure dans le jeu.

1. Instruction du jeu

Avant de commencer la partie, une petite mise en situation ainsi que les instructions de jeu sont présentées au joueur. Ces instructions seront incluses dans la scène d'introduction ou encore, au tout début de votre scène de jeu sous forme de didacticiel.

2. Caméras : Création et utilisation appropriée

Il peut s'agir de différentes caméras permettant de voir l'avatar du joueur ou encore de caméras utilisées pour montrer certaines sections du jeu. (Ex. le personnage ramasse une clé → une caméra s'active temporairement et montre une porte qui se trouve plus loin dans le jeu). Le choix du nombre de camera est laissé à vous.

3. Sons

- Utilisation d'une musique (son 2D ou 3D) pour la scène d'introduction et/ou de conclusion. Utilisez la musique de votre choix. [Audacity](#) est un bon logiciel pour monter vos propres musiques.
- Utilisation d'une ambiance sonore (son 2D) qui devra jouer en continu dès le lancement d'une partie. L'ambiance sonore pourra être réalisée dans le cours de conception sonore.
- Utilisation d'au moins trois sons 3D ponctuels.

4. Éclairage

C'est laisser à votre appréciation. Utilisez les types de lumière que vous voulez (directionnelle et/ou autre).

5. Créations d'objets dynamiques

- Votre jeu devra obligatoirement inclure une création dynamique d'objets.
- La création devra se faire à partir d'un élément préfabriqué (« Prefab »)
- Dans le cas où une multitude d'objets peuvent être créés par l'utilisateur (des balles par exemple), il est nécessaire de gérer efficacement la suppression de ces objets.
- Les objets dynamiques peuvent être des rigidbody, des particules, des ennemis ou d'autres types d'objets.

6. Utilisation de NavMesh (un peu d'IA)

Il faut inclure au moins un agent intelligent dans votre jeu. Si votre concept de jeu n'a pas d'ennemi, alors ça peut être des personnages (ou objet) qui se promènent dans votre monde en suivant des points de destinations choisis au hasard ou d'autres trajectoires déterminées à l'aide de NavMesh. Vous êtes libre pour ce point. Le but est seulement de pouvoir utiliser NavMesh.

Barème de correction :

Description	Pondération
Respect des exigences demandées (utilisation et fonctionnement adéquat)	70
Description, commentaires et organisation des scripts <ul style="list-style-type: none">- une description globale du code avec date et nom et toute information pertinente sur le jeu;- description au début de chaque script ;- description au début de chaque fonction ;- description pour chaque condition ou boucle ;- description des variables dans chaque script, lors de la déclaration;- noms significatifs pour les variables, les scripts et tous les objets présents dans les fenêtres Projet et Hiérarchie ;- organisation des médias dans la fenêtre Projet et des objets dans la fenêtre Hiérarchie. Utilisation de dossiers pour classer les différents éléments.- optimisation de la taille du projet (absence d'éléments (modèles, sons, textures, etc.) non utilisés dans le projet).	10
Créativité dans l'assemblage des interfaces (qualité et aspect artistique, modèles 3D de votre création)	5
Jouabilité (« gameplay ») et appréciation générale	5
Créativité dans la programmation des interfaces <ul style="list-style-type: none">- complexité des scripts (niveau de difficulté qui dépasse celui des exercices ou bonification)- qualité et efficacité des scripts (logique, exécution des scripts, optimisation)	10