Mission X

Date de remise : Point de contrôle : Au plus tard le dimanche 12 novembre 2023 avant 23h59

Remise finale : Au plus tard le mardi 12 décembre 2023 avant 23h59

Modalité de remise : Dans un dossier identifié à votre *nom\_prénom*, en suivant le lien de dépôt qui sera fourni

Point de contrôle et Remise Finale :

* Une version compressée (ZIP) de votre dossier de projet optimisé;  
  (et le package du projet)
* Une version exécutable ou exportée pour le web de votre jeu(WebGL);
* Le plan de votre jeu;

Pondération : 15% au point de contrôle et 30% à la remise finale

Précision sur le TP : Travail individuel

# Ce que vous devez faire - Projet individuel

Dans ce travail pratique, vous devez réaliser un environnement 3D rendu en temps réel à l’aide du logiciel Unity. L’action se déroule à l’intérieur d’un environnement intérieur (ex. un labyrinthe) ou extérieur (terrain) 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. Évidemment, il est impossible pour le joueur de passer à travers les murs ou les différents objets présents dans l’environnement.

Un minimum de cinq objets 3D interactifs devront être mis en place dans l’environnement. Ces objets peuvent avoir une action immédiate ou encore être ramassés par le joueur qui pourra les utiliser plus tard. L’activation des objets peut se faire au simple contact du joueur, mais peut aussi être déclenchée par une touche du clavier ou un autre mécanisme. Dans tous les cas, il sera important de donner une rétroaction à l’utilisateur lorsqu’il activera ou lorsqu’il ramassera un objet. Il peut s’agir d’une rétroaction visuelle ou auditive. La nature des objets ainsi que leur effet dans l’environnement sont laissés à votre discrétion. Pour ne donner que quelques exemples, il peut s’agir d’une potion permettant d’augmenter la vitesse de déplacement de l’utilisateur pendant un certain laps de 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.

Votre jeu doit intégrer l’utilisation d’un compteur. Le compteur peut servir à afficher des points, le nombre d’objets qui restent 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.)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Trois scènes doivent être intégrées au jeu**

**1- Scène d’introduction – avant de commencer la partie**

Avant de commencer une partie, une scène d’introduction (menu) est présentée au joueur. L'interface est générée à l’aide des éléments d’interface utilisateur (UI). Des boutons permettent au joueur de commencer une partie et de consulter le tableau des meilleurs pointages (si présent) . Un champ de texte permet de saisir le nom du joueur. Le nom sera affiché dans la scène de conclusion. Si aucun nom n'est fourni, la partie ne peut pas commencer.

**→N’oubliez pas vos droits d’auteur : mentionnez également votre nom et la date de réalisation !**

**2- Scène jeu**

Le jeu se déroule dans cette scène.

**3- Scène conclusion**

Une fois le temps écoulé ou la mission réussit, une scène de conclusion est chargée. Si le joueur a échoué sa mission, un message (ou une image) signalant son échec s’affiche. N’oubliez pas d’inclure le nom du joueur saisi au départ. Le joueur peut également reprendre la partie du début s’il le désire; la scène du jeu est alors relancée. **Attention ici : si vous avez des variables statiques : vous devez les réinitialiser correctement.**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Autres éléments à inclure dans votre jeu

**Instructions du jeu**

* Avant que la partie ne commence, une petite mise en situation ainsi que les instructions de jeu sont présentées au joueur. Vous pouvez inclure ces instructions dans la scène d’introduction ou encore, au tout début de votre scène de jeu sous forme de didacticiel.

**Caméras**

* Création et utilisation appropriée de trois caméras . 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).

**Sons**

* Utilisation d’une musique (son 2D) pour la scène d’introduction et/ou de conclusion.
* Utilisation d’une ambiance sonore (son 2D) qui devra jouer en continu dès le lancement d’une partie.
* Utilisation d’au moins trois sons 3D ponctuels.

**Éclairage**

* Utilisation appropriée d’au moins deux types de lumière **autre que directionnelle**.

**Système de particules**

* Utilisation appropriée d’au moins deux systèmes de particules ;

**Création dynamique d’objets**

* 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.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Critères d'évaluation: Il est interdit d'emprunter du code sur l'internet ou ailleurs.

**Note importante** : Un plan détaillé (Plan.jpg) de l’environnement devra être remis avec chaque version du TP. Ce plan

devra indiquer la présence d’objets interactifs ainsi que le chemin et les procédures à suivre pour réussir le jeu. Si vous avez plusieurs niveaux , le plan de chaque niveau est nécessaire. Vous pouvez faire la capture d’écran de votre terrain et ajouter les infos dans Photoshop.

## 

**REMISE 1 TP individuel**

| **Description** | **Pondération**  **/10** |
| --- | --- |
| Environnement 3D intégré dans Unity (Qualité de l'assemblage, originalité). (terrain, bâtiment, etc.) | 3 |
| Gestion des caméras (au moins 2 caméras) | 2 |
| Éclairage de l'environnement (au moins 2 types de lumières différentes) | 1 |
| Personnage contrôlable qui ne passe pas à travers les murs et les objets | 2 |
| Présence de deux objets interactifs (2 collisions) | 2 |
| Consignes utiles pour la correction (les contrôles, objectifs, les bogues, etc.)   et/ou plan détaillé du jeu (ex. pour un labyrinthe) | -1 si non remis |

REMISE 2: Version Finale

| **Description** | **Pondération** |
| --- | --- |
| Respect des exigences demandées (utilisation et fonctionnement adéquat) | 70 |
| Description, commentaires et organisation des scripts   * 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, originalité, modèles 3D de votre création, etc.) | 5 |
| Jouabilité (« gameplay ») et appréciation générale | 5 |
| Créativité dans la programmation des interfaces   * complexité des scripts (niveau de difficulté qui dépasse celui des exercices ou bonification) * qualité et efficacité des scripts (logique, exécution des scripts, optimisation) | 10 |

**Note: Pour choisir un personnage pour votre jeu, vous pouvez aller sur le site de Mixamo.com**