



# Piscine Unity - d09

## First Person Shooter

Staff [staff@staff.42.fr](mailto:staff@staff.42.fr)

*Résumé: Ce document contient le sujet du jour 09 de la piscine Unity de [42](#).*

# Table des matières

<b>I</b>	<b>Préambule</b>	<b>2</b>
<b>II</b>	<b>Consignes generales</b>	<b>3</b>
<b>III</b>	<b>Règles spécifiques à la journée</b>	<b>5</b>
<b>IV</b>	<b>Exercice 00 : Arméd and dangerous</b>	<b>6</b>
<b>V</b>	<b>Exercice 01 : This ain't no place for no hero</b>	<b>7</b>
<b>VI</b>	<b>Exercice 02 : Endless Arena</b>	<b>9</b>
<b>VII</b>	<b>Exercice 03 : Who's the boss</b>	<b>10</b>

# Chapitre I

## Préambule

Et oui, c'est déjà presque la fin de la piscine. J'espère que vous avez appris plein de nouvelles choses et que vous avez réussi à faire plein de jeux cools. Pas de vidéos pour aujourd'hui, mais un dernier jeu trop bien avant de terminer en apothéose avec le rush ! Bon courage à tous !

# Chapitre II

## Consignes generales

- La piscine Unity est à faire entièrement et obligatoirement en **C#** uniquement. Pas de Javascript/Unityscript, de Boo ou autres horreurs.
- L'utilisation de fonctions ou de namespaces non autorises explicitement dans le header des exercices ou dans les regles de la journee sera considéré comme de la triche.
- Pour une utilisation optimale de Unity, vous devez travailler sur le ~/goinfre, qui est en local sur le mac que vous utilisez. Pensez à bien récupérer vos projets avant de vous delog car le goinfre local est vidé régulièrement.
- Contrairement aux autres piscines, chaque journée ne demande pas un dossier ex00/, ex01/, ..., exn/. A la place pour la piscine Unity, vous devrez rendre votre dossier projet qui aura pour nom le nom de la journee : d00/, d01/, .... Toutefois, un dossier de projet contient par default un sous-dossiers inutile : le sous-dossier "projet/Temp/". Assurez-vous de ne **JAMAIS** pusher ce dossier dans votre rendu.
- Au cas ou vous vous poseriez la question, il n'y a pas de norme imposée à 42 pour le **C#** pendant cette piscine Unity. Vous pouvez utiliser le style qui vous plaît sans restriction. Mais rappelez-vous qu'un code que votre peer-evaluateur ne peut pas lire est un code qu'elle ou il ne peut noter.
- Vous devez trier les assets de votre projet par dossier. Chaque dossier correspond à un et un seul type d'asset. Par exemple : "Scripts/", "Scenes/", "Sprites/", "Prefabs/", "Sounds/", "Models/", ...
- Assurez-vous de tester attentivement les prototypes fournis chaque jour. Ils vous aideront beaucoup dans la compréhension du sujet et du travail attendu.
- L'utilisation de l'Asset Store d'Unity est interdite. Vous êtes encouragés à utiliser les assets fournis chaque jour (quand nécessaire) ou à en chercher d'autres sur le net s'ils ne vous plaisent pas, sauf bien entendu pour les scripts car vous devez avoir écrit tout ce que vous rendez (hors scripts fournis par le staff, obviously). L'Asset Store est interdit car quasiment tout le travail que vous avez à faire s'y trouve déjà sous une forme ou sous une autre. Néanmoins l'utilisation des Standard Assets de Unity est autorisée voir meme conseillée pour certains exercices.

- Pour les corrections à partir du d03 il vous sera demandé de build les jeux pour les tester. **C'est le correcteur** qui doit build le jeu vous devez donc évidemment toujours push vos projets/sources. De ce fait votre projet doit correctement configuré pour le build. **Aucun** réglage de dernière minute ne doit être toléré.
- Important : Vous ne serez pas évalués par un programme, sauf si le contraire est explicite dans le sujet. Cela implique donc un certain degré de liberté dans la façon que vous choisissiez de faire les exercices. Toutefois, gardez en tête les consignes de chaque exercice, et ne soyez pas FAINÉANTS, vous passeriez à côté de beaucoup de choses intéressantes.
- Ce n'est pas grave d'avoir des fichiers supplémentaires ou inutiles dans votre dossier de rendu. Vous pouvez choisir de séparer votre code en différents fichiers au lieu d'un seul, sauf si le header d'un exercice mentionne explicitement les fichiers à rendre. Un fichier ne doit définir qu'un et un seul comportement, pas de namespaces donc. Toute cette consigne ne s'applique bien évidemment pas au sous-dossier "projet/Temp/" qui n'a pas le droit d'exister dans vos rendus.
- Lisez le sujet en entier avant de commencer. Vraiment, faites-le.
- Le sujet pourra être modifié jusqu'à 4h avant le rendu.
- Même si le sujet d'un exercice est relativement court, ça vaut le coup de passer un peu de temps à comprendre parfaitement le travail attendu pour le faire au mieux.
- Parfois il vous sera demandé un soin particulier sur la qualité artistique de votre rendu. Dans ce cas, cela sera mentionné explicitement dans le sujet correspondant. N'hésitez alors pas à tester plein de choses différentes pour vous donner une idée des possibilités offertes par Unity.
- Par Odin, par Thor ! Réfléchissez !!!
-


# Chapitre III

## Règles spécifiques à la journée

- Pour gérer les mouvements du personnage ne perdez pas de temps et importez directement le First Person Controller du package Characters de Unity qui se trouve dans les Standard Assets.

# Chapitre IV

## Exercice 00 : Arméd and dangerous

	Exercice : 00
Exercice 00 : Arméd and dangerous	
Dossier de rendu : <i>ex00/</i>	
Fichiers à rendre : <b>Tout ce qui vous semble pertinent</b>	
Fonctions interdites : <b>Aucune</b>	
Remarques : <b>n/a</b>	

La base d'un shooter FPS qui se respecte c'est évidemment les armes. Vous allez donc commencer par implémenter ces dernières. Pour cela vous devrez prendre en considération les règles suivantes :


- On peut distinguer deux armes différentes. L'une tire avec précision et ne touchera toujours qu'un seul ennemi. La deuxième fait des dégats de zone et va toucher plusieurs ennemis dans un rayon donné.
- Les armes ont chacune une vitesse d'attaque et des dégats différents.
- Lorsque l'on tire on peut voir le tracé de notre tir et une particule est déclenchée au niveau de l'endroit où a touché notre balle. Là encore on doit pouvoir les distinguer et pouvoir différencier leur rayons/particules.
- On peut passer d'une arme équipée à une autre avec les touches 1 et 2. Et on peut évidemment tirer avec le clic gauche de la souris.
- Les armes produisent un son différent lorsque l'on tire avec et sont animées un minimum (voir la démo du jour).



Si vous ne parvenez pas à finir le deuxième exercice mettez en place des cibles fixes pour tester le fonctionnement des armes. On doit pouvoir leur enlever de la "vie"

# Chapitre V

## Exercice 01 : This ain't no place for no hero

	Exercice : 01
Exercice 01 : This ain't no place for no hero	
Dossier de rendu : <i>ex01/</i>	
Fichiers à rendre : <b>Tout ce qui vous semble pertinent</b>	
Fonctions interdites : <b>Aucune</b>	
Remarques : <b>n/a</b>	

Super on a des armes, maintenant faut degommer des choses sinon on s'ennuie très vite. Ajoutez donc des ennemis doté d'un minimum d'intelligence de sorte à ce que le combat soit intéressant.

Encore une fois voici les consignes minimum pour l'IA vous êtes libre de rajouter ce qu'il vous plaira pour améliorer votre IA :

- Un ennemi part d'un des coins de la map et a pour destination son centre.
- L'IA possède une zone de détection relativement grande. Si le joueur rentre dedans il est poursuivi. Si l'ennemi perd la trace du joueur, il se rend à sa dernière position connue.
- Si l'ennemi est touché il se rend à la position du joueur au moment de son tir. Lorsque la vie de l'ennemi est à 0 il meurt et disparaît.
- Une fois que l'ennemi est suffisamment proche du joueur il l'attaque et lui faire perdre X points de vie par coup.
- L'ennemi doit toujours emprunter le chemin le plus court pour rejoindre sa destination.
- L'ennemi est entièrement animé que ce soit pour lorsqu'il court ou marche mais aussi pour lorsqu'il attaque ou qu'il prend des dégats/meurt.


Disposez quelques ennemis dans une map afin de pouvoir s'amuser dans un cadre



sympa.

# Chapitre VI

## Exercice 02 : Endless Arena

	Exercice : 02
Exercice 02 : Endless Arena	
Dossier de rendu : <i>ex02/</i>	
Fichiers à rendre : <b>Tout ce qui vous semble pertinent</b>	
Fonctions interdites : <b>Aucune</b>	
Remarques : <b>n/a</b>	

Il est temps d'ajouter un gameplay à notre jeu. On va mettre en place un système relativement simple mais qui a fait ses preuves à de nombreuses reprises : l'arène.


Pour que ce soit efficace il va falloir mettre en place plusieurs choses essentielles. Tout d'abord vous allez créer des spawners aux 4 coins de la map. Ils feront apparaître des ennemis dans un laps de temps choisi aléatoirement et de manière non simultanée. Il ne doit jamais y avoir plus de 20 ennemis présents sur la map.

Il va également falloir concevoir un système de vagues. A chaque vague les ennemis deviennent plus forts. Ajoutez donc un facteur progression qui définira la progression globale de la difficulté dans votre jeu. Essayez de calibrer ce dernier pour que votre jeu soit toujours jouable après 5 manches. Prévoyez aussi un petit temps de pause entre deux vagues pendant lequel les ennemis n'apparaissent plus. Les vagues ont un temps défini en revanche même si la vague est terminée les ennemis restent vivants.

On va en profiter pour créer un GUI qui affichera la vie du joueur, le temps restant pour la vague/pause actuelle et qui prévendra du début d'une nouvelle vague/pause par un message. Lorsque l'on meurt le GUI affichera un message et nous donnera le nombre de manche que l'on a passé.

# Chapitre VII

## Exercice 03 : Who's the boss

	Exercice : 03
Exercice 03 : Who's the boss	
Dossier de rendu : <i>ex03/</i>	
Fichiers à rendre : <b>Tout ce qui vous semble pertinent</b>	
Fonctions interdites : <b>Aucune</b>	
Remarques : <b>n/a</b>	

La cerise sur le gateau de votre jeu ca sera donc son boss. Ajoutez donc un ennemi physiquement différent des autres (on le reconnait de loin) qui est doté d'une intelligence artificielle plus évoluée.

Le boss sait toujours où se trouve le joueur et sera toujours à sa poursuite peut importe sa position. Il est doté de beaucoup plus de points de vie et inflige bien plus de dégats. Afin de pimenter un peu le tout le boss sera capable de jeter des boules d'énergie en direction du joueur (pas d'autoguidage cependant, voir la démo). Pour représenter ces boules utilisez une particule, si la boule touche le joueur il est bloqué pendant 2s.

Mais à capacités exceptionnelles, présence exceptionnelle. Ainsi vous créerez donc une vague spéciale Boss intervenant toutes les 3 manches. Cette vague contrairement aux autres ne se termine que lorsque le boss n'a plus de vie. Les ennemis classiques ne spawnent pas durant cette vague.

Comme toujours si vous arrivé jusqu'à là - déjà bravo - mais surtout n'hésitez pas à vous faire plaisir et à ajouter pleins de petits détails sympas pour rendre plus unique et personnel votre jeu !