



# Formation Unity initiation développement jeu vidéo

Bases programmation, programmation orientée objet, méthodologie, langage C#, game design, Unity : concepts, ressources, moteur physique, collisions, interactions, création UI, animations, animator, gestion entrées, exécutables.

**DATES** du 14/11/2022 au 02/12/2022 - du 06/02/2023 au 24/02/2023 - du 22/05/2023 au 09/06/2023 - du 20/11/2023 au 08/12/2023 -  
**DURÉE TOTALE** 15J / 105h

**MODALITES D'ORGANISATION** Formation présentielle, en nos locaux, 13 rue Desargues, 75011 Paris.

**HORAIRES** 10h-13h puis 14h-18h

**NOMBRE DE PARTICIPANTS** 6 maximum

**PARTICIPANTS** Toute personne ayant besoin de s'initier à Unity, et surtout au développement de jeux vidéos.

**NIVEAU REQUIS** Aisance avec les outils informatiques. Niveau maths collège. Compréhension de l'anglais. Entretien pédagogique.

**ACCESSIBILITÉ** Nos formations sont accessibles et aménageables pour les personnes en situation de handicap. Fauteuils roulants autorisés en largeur maxi 70cm. Pour tout handicap, auditif, visuel ou autre, merci de nous contacter pour les aménagements possibles.

**VALIDATION DES ACQUIS** Contrôle continu tout au long de la formation, exercé par le formateur et supervisé par la responsable pédagogique. Délivrance d'un Certificat de Validation des Acquis de Formation.

**MOYENS TECHNIQUES** 6 stagiaires maximum. En présentiel : une station par stagiaire, un iMac récent avec écran 27 pouces et 64Go de Ram (Vram 8 ou 16 Go), Unity, Adobe CC (Photoshop,...), Cinema 4D et Blender, tablette graphique. Accès internet ligne dédiée. En distanciel : un ordinateur pouvant faire tourner Unity et Adobe CC, une connexion internet minimum Adsl, un micro et si possible une webcam (ou un smartphone ou tablette). Nous vous aidons à distance à installer applications, licences, médias et outil de web conference live. En cas de machine insuffisamment performante pour le travail en formation, nous vous mettrons gratuitement à disposition une station à distance (cloud computing) ou un de nos ordinateurs (selon disponibilité).

**MOYENS ET MODALITES PEDAGOGIQUES** Une salle de formation avec tableau blanc, un poste informatique pour chaque apprenant, un poste informatique vidéo-projeté pour les formateurs, avec accès internet et imprimante partagée. Alternance de séquences théoriques et d'exercices d'application réelle. Les exercices sont réalisés sous la supervision des formateurs, dans une logique d'apprentissage des compétences et d'autonomie progressive des apprenants. Des exercices plus longs et synthétiques permettent de reprendre un ensemble de compétences dans des cas pratiques types des métiers et compétences concernés, et s'assurer de leur acquisition par les apprenants.

**SUPPORT DE COURS** 1 mémo et 1 livre

**FORMATEURS** Professionnel en activité, expert reconnu dans son domaine, animant régulièrement des formations.

**LABELS QUALITÉS** Centre certifié Qualiopi (certification nationale) / certifié ISQ-OPQF / référencé Data-Dock / centre certifié Adobe / Maxon / Blackmagic Design / centre de test Certiport

## OBJECTIFS DE LA FORMATION

**Ce stage de formation Unity d'une durée de 3 semaines est une formation d'initiation complète et intensive au moteur temps réel Unity pour des applications en jeu vidéo 2D.**

Cette formation de 3 semaines sera intense : il s'agira d'acquérir des concepts de programmation informatique et de les fusionner avec des éléments audiovisuels, c'est pour cela que cette formation se limitera au développement de jeu vidéo en 2D. A la fin de la formation les stagiaires seront autonomes pour réaliser des prototypes de jeux vidéos et se sentir suffisamment à l'aise pour progresser par eux-mêmes.

#### **Objectifs de la formation Unity initiation :**

- Comprendre le flux de production et l'ergonomie de Unity
- Maîtriser les concepts majeurs : *gameobjects*, *components* et boucle de rendu
- Importer des ressources différentes (images, sons, animations...)
- Utiliser le moteur de physique (simulation de la gravité, des chocs et des matériaux)
- Gérer des collisions, des *triggers* et des *layer-masks*
- Utiliser et créer des petits programmes (en C#) pour interagir avec les éléments du jeu et changer leur comportement
- Créer une interface utilisateur interactive
- Créer des menus
- Sauvegarder des données du joueur
- Créer de niveaux de jeu avec les outils 2D (*TileMap*, *SpriteShape*...)
- Animer des éléments avec des *sprite sheets*, des images clés ou du code
- Gérer des animations avec le module *Animator* pour les lancer à des moments opportuns
- Récupérer les différentes entrées (clavier, souris, manette...) pour interagir avec le personnage
- Créer un fichier exécutable pour un site internet, un ordinateur ou un *smartphone*

## **CONTENU COMPLET DE LA FORMATION**

*Ce stage peut s'aborder de deux façons différentes : apprendre à coder par l'entremise du jeu vidéo (ce qui est nettement plus amusant que de coder une application de gestion de flux maritime), ou bien, apprendre à se servir du moteur de jeux Unity et donc, savoir un peu coder.*

#### **CONCEPTS DE PROGRAMMATION**

*Basiquement, un ordinateur ne sait faire que 4 choses : mettre en mémoire des nombres, faire des opérations sur ces nombres, faire une action plutôt qu'une autre en fonction après évaluation du résultat, et répéter joyeusement tout ça tant qu'on ne lui dit pas d'arrêter.*

#### **La formation permettra de comprendre ces grands principes de bases de la programmation :**

- les variables : comment un programme met en mémoire la valeur d'une donnée et peut l'utiliser de nouveau ?
- les conditions : comment un programme utilise la valeur d'une variable pour lancer telle ou telle action ?
- les boucles : comment demander au programme de répéter une action et surtout de l'arrêter à un moment ?
- les fonctions : comment se faciliter la vie pour ne pas à avoir à recopier des parties du code dont on se sert plusieurs fois ?

#### **PROGRAMMATION ORIENTEE OBJET**

Mais avec le temps, les informaticiens se sont aperçus que d'ordonner ces 4 processus de bases dans des algorithmes toujours plus complexes rendaient les programmes trop compliqués à écrire, à maintenir et à exécuter. Il fallait aussi structurer les données que l'on entrait dans le programme. Ils ont alors créé le concept puissant bien qu'un peu abscons, d'Objet.

**Nous aborderons donc la programmation orientée objet et nous découvrirons avec bonheur sa puissance au travers de certains principes :**

- l'encapsulation : comment empêcher le programme de faire n'importe quoi avec les données.
- l'héritage : comment éviter de recréer la roue à chaque fois en se servant de choses que l'on a déjà créées.
- les classes : comment ordonner nos données pour qu'elles puissent représenter tout ce que l'on désire modéliser en leur donnant des comportements et des attributs.

## LANGAGE C# ET METHODOLOGIE

**Unity utilise le langage C#** (prononcé « C Sharp ») qui a l'avantage d'être un langage moderne, à la syntaxe relativement simple et proche d'autres langages très utilisés comme le Java (le langage le plus utilisé actuellement) ou Swift (le langage utilisé pour créer des applications sur les appareils Apple).

Cependant, empiler des lignes de codes sans méthode rend le développement d'un jeu et sa maintenance très difficiles voire impossibles. Pour cela, une petite initiation à certaines architectures logicielles simples sera prodiguée avec bienveillance :

- Minimiser le couplage et le partage de données entre les différents éléments du jeu par l'intermédiaire d'un *game manager*,
- Utilisation de *State machine* pour animer un personnage,  
Gestion de la mémoire et des rayons laser avec l'*object pooling*

## UNITY

Unity est un moteur de jeux assez populaire, gratuit (tant que vous ne gagnez pas beaucoup d'argent) et, au final, assez facile à utiliser. Il sert, bien entendu, à faire des jeux mais on le retrouve aussi dans l'architecture, le design industriel, et l'animation 3D. Unity fonctionne comme un grand intégrateur qui va mettre en relation différents éléments audiovisuels créés avec des logiciels comme Photoshop, Illustrator, Cinema4D, Blender...

- Apprentissage du flux de production et de l'ergonomie de Unity
- Apprentissage des concepts majeurs : *gameobjects*, les *components* et la boucle de rendu
- Importations de ressources différentes (images, sons, animations...)
- Utilisation du moteur de physique (simulation de la gravité, des chocs et des matériaux)
- Gestion des collisions, des *triggers* et des *layer-masks*
- Utilisation et création de petits programmes (en C#) pour interagir avec les éléments du jeu et changer leur comportement
- Création d'une interface utilisateur interactive
- Animations d'éléments avec des *sprite sheets*, des images clés ou du code
- Gestion des animations avec le module *Animator* pour les lancer à des moments opportuns
- Récupérer les différentes entrées (clavier, souris, manette...) pour interagir avec le personnage
- Créer un fichier exécutable pour un site internet, un ordinateur ou un *smartphone*

## GAME DESIGN

*Il existe de longs débats pour savoir si le jeu vidéo peut-être considéré comme de l'Art. En revanche, si ça n'est pas de l'Art c'est tout un art que de faire un jeu vidéo.*

Nous aborderons donc, aussi, dans cette formation quelques concepts fondamentaux du *Game Design* :

- Définir les différentes boucles de *gameplay* qui permettent d'intéresser les joueurs et joueuses
- Définir les 3C d'un jeu : *Character, Control, Camera* pour que le public éprouve l'expérience souhaitée
- Définir les récompenses et les feedbacks pour guider les joueurs et joueuses
- Ajouter du *juice* pour rendre le tout *fun* et addictif

### **METHODOLOGIE PEDAGOGIQUE, PRATIQUE**

Même s'il faudra de temps en temps passer par un peu de théorie, la formation sera essentiellement constituée d'exercices pratiques, pour comprendre, appliquer, mémoriser, tester, et devenir réellement autonome.

Tout au long de ces 3 semaines, vous créerez différents jeux s'inspirant des genres les plus emblématiques du jeu vidéo 2D (*runner, plateformer, shoot'em up, aventure en top-down...*) afin de mettre en pratique tous les concepts abordés.