

Formation Unity : UX pour les applications XR

Durée :	5 jours
Public :	UX/UI designer, développeur d'applications XR, développeur de jeux vidéo
Pré-requis :	Notions de bases en programmation orientée objet, maîtrise d'un logiciel de création graphique, notions en animation.
Objectifs :	Concevoir des expériences utilisateur et des interfaces graphiques - Développer des applications XR
Sanction :	Attestation de fin de stage mentionnant le résultat des acquis
Taux de retour à l'emploi:	Aucune donnée disponible
Référence:	CAO101371-F
Note de satisfaction des participants:	Pas de données disponibles

Introduction

Définition de la XR : historique, utilisation, propriétés, références.
Présentation et analyse du fonctionnement d'applications VR, AR et MR.

Notions fondamentales

Les possibilités et contraintes des applications en Réalité Virtuelle, Augmentée ou Mixte.
Analyser les contraintes matérielles.
Comprendre les éventuelles problématiques liées à l'utilisation de matériel XR (Motion sickness, désorientation, etc)

Conception d'expériences et d'interfaces utilisateurs XR

Analyser les différences entre la 3D classique, la Virtual Reality, l'Augmented Reality et la Mixed Reality.
Préparer des expériences d'utilisation d'application dans le domaine de la XR.
Concevoir des interfaces utilisateurs (graphiques, audio, tactiles, etc) dans une application XR.

Présentation de l'interface de Unity

L'éditeur (Barre de menu, les vues Scène et Jeu, les panneaux : Projet, Hiérarchie, Inspector)
L'éditeur de code associé : Visual Studio Community
La console de débogage
La documentation en ligne

La création de projet 3D avec Unity

Installer les modules spécifiques à la XR via le Unity Hub

Créer et paramétrer un projet VR ou AR

Installer et activer des plugins officiels

Installer le plugin XR Interaction toolkit de Unity

Atelier : créer et paramétrer un projet XR, installer les plugins nécessaires au fonctionnement sur le matériel spécifique

Les modèles 3D

Se déplacer dans une scène 3D, manipuler les différentes vues

Créer et modifier des primitives 3D

Intégrer des modèles 3D dans un projet (fbx, obj, dae, 3ds, dxf, etc)

Atelier : créer une scène 3D avec des primitives et des objets importés.

Les Shaders et les matériaux

Comprendre les différents types de Shaders

Créer un shader à partir du Shader Graph.

Créer des matériaux

Appliquer des textures aux objets 3D

Appliquer shaders et matériaux aux objets 3D présents dans la scène

Atelier : créer différents matériaux et shaders.

Caméra

Comprendre les spécificités des caméra XR.

Effectuer les réglages caméra en adéquation avec les contraintes du projet.

Mettre en place un système de caméra CineMachine.

Atelier A : créer une scène en réalité augmentée

Atelier B : créer une scène en réalité virtuelle

Eclairage

Positionner et manipuler différentes sources lumineuses (Point, Spot, Area, Directionnal)

Mettre en place le pré-calcul des éclairages (bake vs. real time)

Atelier : ajouter différentes sources lumineuses dans une scène, implémenter un cycle jour/nuit.

La physique

Ajouter et régler un composant Rigidbody

Comprendre les différentes méthodes pour détecter les collisions.

Ajouter et régler un collider

Créer et ajuster un matériau physique 3D.

Atelier : appliquer différents colliders et rigidbodies à plusieurs primitives 3d pour créer un jeu basé sur l'utilisation du moteur physique.

Les Prefabs

Découvrir les Prefabs, leurs avantages, leur fonctionnement

Créer des Prefabs dans un projet

Instancier un Prefab et modifier le modèle

Atelier : Créer un Prefab et l'instancier de manière récursive avec un script dédié.

Audio

Gérer le son dans Unity

Importer et organiser des clips audio (formats de fichiers, échantillonnage, mono/stéréo)

Mettre en place des composants audios spécifiques à une application XR

Atelier : déclenchement de sons liées à l'utilisation d'une interface utilisateur VR ou AR.

Scripts (C#)

Comprendre la structure du code dans Unity

Comprendre la boucle de jeu et les méthodes associées : Update, FixedUpdate, LateUpdate

Implémenter des classes héritées de MonoBehaviour

Gérer les événements (Collision, Input, etc)

Créer et implémenter les Actions.

Atelier : Implémenter un système de contrôle utilisateur via la création de scripts.

Présentation du système d'UI

Mettre en place un Canvas

Création d'interfaces graphiques sous forme d'éléments 3D (écran tactile, contrôles en 3d, etc)

Atelier : mettre en place différents systèmes de menu (2D en surcouche ou dans l'environnement 3D)

Les Scènes

Créer plusieurs Scènes pour un même projet

Organiser et déclencher la succession de scènes

Conservé certains paramètres d'une scène à l'autre

Mise en place du système de PlayerPrefs (stockage de données extérieure à l'application)

Atelier : Ajouter des scènes au projet, assurer de bonnes transitions. Mettre en place la sauvegarde perpétuelle de certaines données.

Compilation

Débugger et exécuter

Mettre en place un monitoring des performances

Régler les dépendances et plateformes cibles

Générer les fichiers exécutables (Windows, OSX, Linux, iOS, Android, Oculus)

Atelier : Analyser le manque d'optimisation d'une application, y apporter des corrections. Compiler et déployer l'application sur la machine ou device spécifique.