

Curriculum pour élèves de 4èmes a 2èmes

Mois 1 : Introduction à Unity et Concepts de Base

- Semaine 1-2 : Démarrer avec Unity
 - Introduction à l'interface et à l'espace de travail Unity.
 - Création et configuration de projets Unity.
 - Aperçu des objets de jeu, des composants et de la hiérarchie.
- Semaine 3-4 : Création d'un Simple Jeu 2D
 - Configuration des contrôles et du mouvement du joueur.
 - Implémentation d'interactions de base et de collisions.
 - Introduction à l'animation de sprite.

Mois 2 : Concepts Intermédiaires Unity et Bases du Scripting

- Semaine 5-6 : Comprendre les Scripts dans Unity
 - Introduction au scripting C#.
 - Écriture de scripts pour contrôler le comportement du jeu.
 - Utilisation de variables, fonctions et conditions dans les scripts.
- Semaine 7-8 : Ajout de Plus d'Interactivité
 - Implémentation de mécanismes de jeu comme les systèmes de score et de santé.
 - Gestion des entrées utilisateur et interactions avec l'interface utilisateur (UI).
 - Introduction au son et aux effets sonores.

Mois 3 : Polissage et Raffinement des Compétences en

Développement de Jeux

- Semaine 9-10 : Travail avec les Scènes et les Transitions
 - Création de plusieurs scènes et gestion des transitions.
 - Implémentation du design de niveaux et de la progression.
 - Introduction au contrôle de caméra et aux effets.
- Semaine 11-12 : Optimisation et Performance
 - Compréhension des techniques d'optimisation pour les jeux 2D.
 - Profilage et débogage des problèmes de performance.

- Implémentation des meilleures pratiques pour un développement de jeu efficace.

Mois 4 : Mécaniques de Jeu Avancées et Intégration d'Actifs

- Semaine 13-14 : Physique 2D Avancée
 - Exploration de la détection avancée de collisions et des simulations physiques.
 - Implémentation de mécaniques de jeu comme le saut et la gravité.
- Semaine 15-16 : Intégration d'Actifs de Tiers
 - Importation et intégration d'actifs depuis l'Unity Asset Store.
 - Personnalisation et extension des actifs pour répondre aux besoins du projet.
 - Gestion des dépendances d'actifs et du contrôle de version.

Mois 5 : Développement de Projet et Collaboration

- Semaine 17-18 : Planification de Projet et Prototypage
 - Brainstorming et conceptualisation d'idées de jeu.
 - Création de plans de projet et de calendriers.
 - Prototypage des mécaniques de jeu et des fonctionnalités de base.
- Semaine 19-20 : Développement de Projet en Groupe
 - Travail collaboratif sur un projet de groupe.
 - Attribution des tâches, gestion du flux de travail et communication.
 - Itération sur le projet en fonction des retours et des tests.
 - Introduction à GitHub

Mois 6 : Finalisation de Projet et Présentation

- Semaine 21-22 : Finalisation et Polissage des Projets
 - Polissage des mécaniques de jeu, des visuels et de l'interface utilisateur (UI).
 - Tests et débogage pour la version finale.
- Semaine 23-24 : Présentation et Réflexion
 - Présentation des projets achevés aux pairs et aux mentors.
 - Collecte des retours et réflexions sur le parcours d'apprentissage.
 - Célébration des réalisations et des prochaines étapes dans le développement de jeux.

Curriculum pour élèves de première à Terminales

Bases de Linux (4 semaines)

Semaine 1: Introduction à Linux

- Présentation du système d'exploitation Linux et de ses distributions.
- Utilisation de la ligne de commande : commandes de base et navigation.
- Gestion des fichiers et des répertoires.

Semaine 2: Gestion des Processus et des Utilisateurs

- Gestion des processus : affichage, arrêt, et priorités.
- Administration des utilisateurs : création, suppression et gestion des permissions.
- Introduction aux scripts shell.

Semaine 3: Réseau et Sécurité

- Configuration réseau : connexion, adresse IP, et pare-feu.
- Gestion de la sécurité : permissions, cryptage, et gestion des paquets.
- Utilisation de SSH pour l'accès à distance.

Semaine 4: Automatisation et Maintenance

- Automatisation des tâches avec les scripts shell et les tâches planifiées.
- Maintenance système : mises à jour, sauvegardes, et dépannage.
- Gestion des services système et des journaux.

Introduction à React avec JavaScript et HTML (8 semaines)

Semaine 5-6: Introduction à JavaScript

- Fondamentaux de JavaScript : variables, types de données, et opérateurs.
- Structures de contrôle : boucles, conditions, et fonctions.
- Introduction au DOM (Document Object Model).

Semaine 7-8: HTML et CSS

- Introduction à HTML : balises, éléments, et structure de page.
- Mise en forme des pages avec CSS : styles, sélecteurs, et mise en page.
- Création de pages web statiques.

Semaine 9-10: Fondamentaux de React

- Présentation de React et de ses concepts fondamentaux.
- Création de composants React : classe et fonctionnels.
- Utilisation de JSX pour incorporer JavaScript dans HTML.

Semaine 11-12: Gestion de l'État et des Événements

- Gestion de l'état local et global dans React.
- Manipulation des événements utilisateur.
- Introduction aux Hooks React.

Apprentissage du Langage C (6 semaines)

Semaine 13-14: Bases du Langage C

- Introduction au langage de programmation C.
- Variables, types de données, et opérateurs.
- Structures de contrôle : boucles, conditions, et fonctions.

Semaine 15-16: Tableaux et Pointeurs

- Utilisation des tableaux et des chaînes de caractères.
- Manipulation des pointeurs et allocation dynamique de la mémoire.
- Structures de données simples : listes et piles.

Semaine 17-18: Fonctions et Fichiers

- Fonctions : déclaration, définition, et appels.
- Manipulation des fichiers : ouverture, écriture, et lecture.
- Gestion des erreurs et débogage.

Raspberry Pi et IoT (6 semaines)

Semaine 19-20: Introduction à Raspberry Pi

- Configuration initiale de Raspberry Pi et mise en place de l'environnement de développement.
- Interfaçage avec les composants matériels : LED, capteurs, et boutons.
- Introduction aux projets IoT et à la programmation GPIO.

Semaine 21-22: Projets Pratiques avec Raspberry Pi

- Réalisation de projets pratiques : surveillance de la température, automatisation domestique, etc.
- Utilisation de protocoles de communication : I2C, SPI, et UART.
- Développement de projets IoT basés sur le cloud.

Semaine 23-24: Finalisation de Projets et Présentation

- Finalisation des projets en cours et tests.
- Présentation des projets aux pairs et aux mentors.
- Réflexion sur le parcours d'apprentissage et les prochaines étapes.

