

Enseignant(s)

VIDAL Nicolas

Email(s)

nvidal@myges.fr

Projet Annuel 3A - ESGIBG

1 Matières, formations et groupes

Matière liée au projet : **Programmation Jeux Vidéos**

Formations : **3ESGI - Groupe 3DJV ||**

Nombre d'étudiant
par groupe :

3

Règles de constitution des groupes: **Imposé**

Charge de travail
estimée par étudiant : **50,00 h**

2 Sujet(s) du projet

Type de sujet : **Imposé**

3 Détails du projet

Objectif du projet (à la fin du projet les étudiants sauront réaliser un...)

Réaliser un premier Jeu Vidéo complet multijoueur, en utilisant au maximum l'ensemble des outils mis à disposition par l'environnement de développement de jeu vidéo Unity avec l'objectif de pouvoir réutiliser la majeure partie des assets développés pour une éventuelle production plus sérieuse (pipeline d'intégration générique, code réutilisable, prefabs génériques, ...).

Descriptif détaillé

Réaliser un jeu vidéo multijoueur intégrant se basant sur le principe d'une Battle Royale.

Contraintes

Type de Jeu

L'idée est de laisser une liberté conséquente à l'équipe de projet pour permettre de valoriser une création dans le cadre de leur scolarité, tout en ayant un socle commun permettant une évaluation sensée des différents projets les uns par rapport aux autres.

Innovation

L'originalité sera appréciée, notamment dans le but de ne pas voir uniquement des clones de PUBG ou de Fortnite.

Il sera attendu de la part du groupe d'étudiant de pouvoir justifier la totalité des choix d'implémentations et d'architecture. Il sera important de réfléchir longuement avant de se lancer dans l'écriture du code.

Gameplay

Le gameplay doit s'adapter aux contraintes suivantes :

? Multijoueur

? Gestion d'inventaire in-game et consommables

? Environnement 3D

? Une partie de l'environnement doit être générée à chaque partie (items/bâtiments, etc.)

Challenges Techniques (1 au choix)

? Environnement démesuré : La taille de l'environnement de jeu devra être démesuré, et poser un réel problème technique pour être chargé en mémoire en intégralité sur la plateforme cible

? Bots non naïfs : Le/les joueurs devront pouvoir se mesurer à des avatars contrôlés par le moteur de jeu ou un client automatisant leurs actions. Ces derniers devront être crédibles. L'utilisation des Unity ML Agents est envisageable, mais attention, grande difficulté ici selon le type de jeu.

? Cutsscenes et objectifs intermédiaires : Pour gagner la partie les joueurs doivent être amenés à réaliser de nombreux objectifs intermédiaires, ces derniers doivent déclencher des cutsscenes animées pour TOUS les joueurs connectés et ainsi mettre en pause temporairement le déroulé du jeu avant de laisser le jeu reprendre son cours.

? Mobile Friendly : le jeu devra pouvoir fonctionner en 4g entre plusieurs téléphones Android.

? Editeur de carte inclus au produit: Le joueur pourra éditer, stocker (en ligne ou sur son device) des environnements de jeu, et les proposer au démarrage de la partie (écran de lobby)

Challenges Technologiques (1 au choix)

? Réseaux Sociaux : Le joueur sera incité à des moments opportuns à partager ses exploits sur différents réseaux sociaux (Facebook, Twitter et Instagram), seule l'utilisation des frameworks et plugins officiels sera autorisée ici.

? Stockage en ligne du profil de joueur et ranking : Le joueur devra pouvoir créer un compte utilisateur et stocker ses informations sur un serveur distant, un site web associé devra être développé pour afficher les classements, exploits des utilisateurs (en plus de pouvoir directement visualiser ceux-ci dans le jeu)

? Stockage en ligne du profil de joueur, et boutique : Le joueur devra pouvoir obtenir une monnaie virtuelle tout en faisant des parties, et cette monnaie lui permettre de débloquent des objets/compétences/... entre les parties. Une interface d'administration (web) permettant d'ajouter des objets devra également être développée pour pouvoir mettre à jour cette boutique sans avoir à mettre en ligne une nouvelle version du jeu.

? Publicités : Le jeu devra intégrer un framework publicitaire, et l'utiliser pour proposer des bannières, des interstitiels et des rewarded videos. Par exemple ces dernières peuvent permettre de débloquent des items ou de commencer le jeu avec de meilleures statistiques.

Features obligatoires

? Menu, options, etc. Le jeu devra contenir au minimum :

o Un Ecran titre

o Un écran de lobby : Espace d'attente de connexion à la partie, le temps que tous les joueurs soient prêts.

o Un écran d'options permettant :

- De désactiver les effets sonores et/ou la musique (ou régler différents volumes sonores)

- De changer la langue de manière dynamique

- De régler la qualité visuelle du jeu

o Hud InGame permettant de mettre le jeu en pause, et de quitter le jeu en revenant au menu principal

o Un écran de fin de partie résumant les statistiques et résultats de la partie

? Localisation

o Le jeu devra être disponible en Anglais et Français, changer de langue devra être possible sans avoir à

redémarrer le jeu depuis le menu d'options.

? Robustesse

- o Mise en pause in-game Multijoueur (tous les joueurs doivent se retrouver avec le menu pause avec un vote de reprise proposé)
- o Indépendance à la résolution d'écran
- o Indépendance au format d'écran

? Performances

- o 60fps constant sur la plateforme cible

Pour permettre au jury de pouvoir évaluer correctement le jeu produit il sera nécessaire de montrer en peu de temps le déroulé d'une partie. Trois options s'offrent à vous :

? Vous baser sur un format court de partie (5-8 minutes)

? Proposer un mécanisme pour avancer dans la partie plus rapidement lors de la soutenance (sauvegarde/chargement in-game)

? Cheat codes

Plateformes Cibles

La plateforme cible : Windows (ou Android si challenge Technique choisi)

Programmation d'outils

En dehors du rendu du jeu proprement dit, il sera demandé le développement d'un minimum de 2 outils (pouvant prendre la forme de scripts/plugins pour Unity ou de programmes complètement indépendants) ayant pour but de faciliter la conception du jeu.

Ex : génération automatique de niveau, génération de scènes de test, visualisation de l'équilibre des factions si plusieurs, etc.

Outils

Moteur de Jeu

L'utilisation d'Unity comme moteur/environnement de développement est requise.

Scripting

IDE

L'utilisation de JetBrains Rider ou Microsoft Visual Studio ou Monodevelop est vivement conseillée, de même que l'utilisation du langage C# pour le scripting, il sera également possible d'avoir recours à l'utilisation de plugins natifs dans le but d'obtenir de meilleures performances.

Coding Style

Tous les scripts développés devront respecter un coding style identique. Il est envisageable de reprendre des coding styles classiques et existants tels que :

? http://www.codeproject.com/KB/cs/c__coding_standards.aspx

? <http://csharpguidelines.codeplex.com/releases/view/46280>

Dans tous les cas, le code produit devra être propre (notamment exempt de traces de debug), et commenté.

Versionning

L'utilisation de Git SCM est obligatoire. De même, il est vivement conseillé de produire régulièrement des archives de l'intégralité du projet en cours et de dupliquer les supports de sauvegardes dans le but d'éviter les traditionnelles pannes de disques durs au moment des soutenances.

Sources

L'intégralité des sources devront être accessibles au moment du rendu final et à chaque jalon dans le but de pouvoir contrôler la bonne application des contraintes présentes dans ce document.

Rendu

Chaque rendu sera effectué via le pull de la branche master du dépôt Git par l'enseignant. Tout retard et/ou problème sera sanctionné par une perte de points. Une archive Zip des builds du jeu et des sources correspondant à chaque jalon sera également uploadée sur MyGES et ces derniers seront également disponibles sur Unity Cloud Build.

Valorisation de l'enseignement

L'intégralité des enseignements ayant traités à la production d'un jeu vidéo devront être mis en avant dans ce projet selon leur applicabilité lors des différents jalons (GameDesign, Modélisation3D, Gestion de

projet, ...).

Par exemple, les étudiants seront tenus de faire figurer leurs productions 3D issues de leur cours de modélisation/animation dans le produit multimédia résultant.

Authoring

Dans le but de pouvoir clairement identifier le travail de chacun au sein du groupe de projet, tout fichier produit devra porter soit dans son nom, soit dans son entête, le nom de son auteur ainsi que son numéro de version. Si cela se révèle être problématique pour le nommage de certains assets, il est aussi envisageable de créer un fichier texte récapitulatif, décrivant clairement pour chaque fichier produit la contribution de chacun.

Réutilisation de code / bibliothèque existant

A priori, aucune utilisation de bibliothèque existante en dehors de l'API de U

Ouvrages de référence (livres, articles, revues, sites web...)

<http://unity3d.com/support/documentation/> ; <http://fr.openclassrooms.com/informatique/cours/realisez-votre-premier-jeu-video-avec-unity> ;

Outils informatiques à installer

Unity(développement) <http://unity3d.com/unity/download> , Suite Office (présentation et document de suivi de projet), Git SCM (source control and versioning) : <http://git-scm.com/>

4 Livrables et étapes de suivi

1	Etape intermédiaire	PROTOTYPE Version comprenant la boucle de gameplay principale du projet Livrables : ? Exécutable PC/Windows ? Sources (répertoire du projet Unity) ? Bref rapport présentant la TODO list	mercredi 20/02/2019 23h59
2	Etape intermédiaire	ALPHA Version comprenant la majorité des éléments de Gameplay du produit multimédia. Livrables : ? Exécutable PC/Windows ? Sources (répertoire du projet Unity) ? Bref rapport présentant la TODO list	jeudi 18/04/2019 23h59
4	Etape intermédiaire	BETA Version quasi-complète, incluant l'intégralité des assets, pouvant posséder quelques bugs mineurs. Livrables : ? Exécutable PC/Windows ? Sources (répertoire du projet Unity) ? Bref rapport présentant la TODO list	dimanche 16/06/2019 23h59

5	Etape intermédiaire	<p>Release Candidate</p> <p>Rendu Packagé, Version du jeu Bug Free, avec installer. Bonus : site web vitrine, notice, jaquette, etc. Livrables : ? Exécutable PC/Windows ? Sources (répertoire du projet Unity) ? Bref rapport présentant l'évolution du projet initial par rapport au produit réalisé</p>	<p>vendredi 19/07/2019 23h59</p>
6	Rendu final	<p>Soutenance</p> <p>Présentation 'commerciale' du jeu devant un public ouvert et hétérogène. Livrables : ? Présentation du projet (Powerpoint) ? Démonstration pertinente et ciblée</p>	<p>samedi 20/07/2019 10h00</p>

5 Soutenance

Durée de présentation par groupe :	30 min	Audience : Publique
Type de présentation :	Présentation / PowerPoint - Démonstration	
Précisions :		