



xV : x facons de visualiser les choses

Projet Unity

ly ly@staff.42.fr
42 Staff pedago@staff.42.fr

Résumé: Construire une application sous Unity permettant à son utilisateur de créer et d'animer un environnement industriel virtuel.

Table des matières

I	Préambule	2
II	Introduction	3
III	Objectifs	4
IV	Consignes générales	5
V	Partie obligatoire	6
V.1	Présentation	6
V.2	Cahier des charges	6
V.3	Images provenant de la POC	8
VI	Partie bonus	10
VII	Rendu et peer-évaluation	11

Chapitre I

Préambule



Ce sujet vous est proposé par l'entreprise Daher. Daher est un équipementier qui développe des systèmes industriels intégrés pour l'aéronautique et les technologies avancées. Tourné vers l'innovation depuis sa création en 1863, Daher s'impose aujourd'hui comme l'un des acteurs majeurs de la 3e révolution industrielle. Ses domaines d'activité sont multiples : construction d'avion, services nucléaires, aérostructures et systèmes, logistique intégrée...

Chapitre II

Introduction

Avez-vous déjà joué aux sims ? Ce jeu un peu nul, sans challenge où on observe des personnages faire des actions de la vie de tous les jours. Une adaptation de ce jeu-là pour un environnement industriel serait pour nous une réelle avancée.

Avec un bâtiment industriel à la place de la maison.

Remplacer la bouffe du frigo par des pièces en caoutchouc.

Pouvoir prendre ces pièces, ne pas pouvoir les manger, mais pouvoir les poser sur une table ou dans une machine.

Faire interagir des objets et des hommes.

C'est ce qu'on veut.

Chapitre III

Objectifs

Nous avons besoin de montrer concrètement ce qu'on fait dans nos usines et nos entrepôts.

Les logiciels de l'industrie sont aujourd'hui beaucoup trop onéreux, non fonctionnels (on ne voit pas un homme faire un mouvement proche de la réalité) et possèdent des décors dignes des plus mauvais jeux de PlayStation 1.

Après une étude du meilleur outil pour notre projet, nous avons choisi la technologie de la 3D temps réelle pour faire une application qui nous permet de représenter notre infrastructure puis d'y intégrer des processus homme/machine.

Une preuve de concept a déjà été réalisée. C'est pour avoir un outil plus robuste que nous nous tournons vers vous, les spécialistes de la programmation.

Chapitre IV

Consignes générales

- Ce projet est un projet Unity, il est donc à faire entièrement et obligatoirement en C# uniquement. Pas de Javascript/Unityscript, de Boo ou autres horreurs.
- Au cas où vous vous poseriez la question, il n'y a pas de norme imposée à 42 pour le C#. Vous pouvez utiliser le style qui vous plaît sans restriction. Mais rappelez-vous qu'un code que votre peer-évaluateur 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/", ...
- L'utilisation de l'Asset Store d'Unity est interdite. Vous êtes encouragés à utiliser les assets fournis 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. Néanmoins l'utilisation des Standard Assets de Unity est autorisée.
- Les corrections se feront en peer-correcting.

Chapitre V

Partie obligatoire

V.1 Présentation

Le périmètre de l'application dans son utilisation comporte :

- La création de l'environnement
- Son paramétrage
- Les actions et animations qu'on souhaite voir
- Un système qui permet de mieux visualiser le tout

Dans notre preuve de concept, nous faisons les choses en 3 étapes. D'une part nous créons l'environnement, d'autre part, nous faisons les animations. Enfin, nous lançons un replay des actions, ce qui génère un mini film à l'écran dans lequel on peut se déplacer.

V.2 Cahier des charges

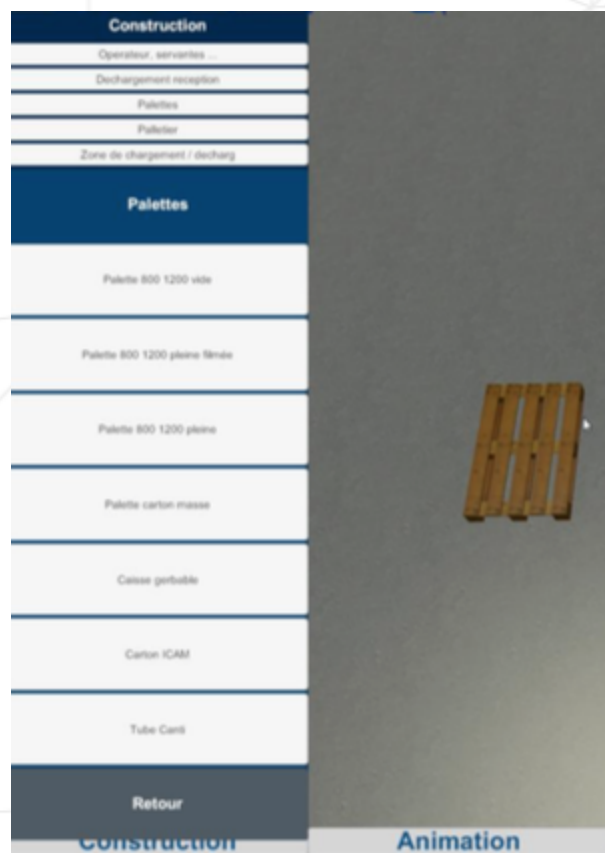
Ainsi, l'application devra comporter plusieurs fonctions :

1. Construire un environnement à partir de modèles 3D :
 - Déjà intégré dans l'application
 - A importer dans l'application et à garder en mémoire
2. Pouvoir modifier cet environnement :
 - Modifier le nom
 - Modifier la position
 - Modifier la rotation
 - Détruire l'objet
 - Modifier la couleur

3. Pouvoir animer cet environnement :
 - Permettre l'interaction entre deux objets
 - Permettre de lancer des mouvements d'objets (animation, translation, rotation)
 - Modifier une séquence
 - Afficher du texte expliquant l'action en cours (l'utilisateur doit pouvoir le changer)
 - Pouvoir paramétrer les actions et les animations (changement des vitesses)
 - Créer des enchaînements d'action : une homme prend une pièce et la pose à un endroit. L'application doit pouvoir générer la tâche en boucle.
4. Pouvoir sauvegarder tout le contenu :
 - Être en mesure de revenir après avoir quitté l'application à l'endroit où nous avons laissé l'application
 - Charger une autre sauvegarde d'un autre environnement
5. Avoir plusieurs modes de vue :
 - Vue d'ensemble
 - Vue 1ère personne
 - Se déplacer librement au sol pour la première personne et dans l'espace pour la troisième personne
6. Avoir un système de replay :
 - Permettre de rejouer des séquences d'animations
7. Exporter une vidéo des séquences animées
8. Rendre l'application ergonomique :
 - Menu
 - Boutons

Nous vous fournirons un set de modèles 3D ainsi que des animations humaines.

V.3 Images provenant de la POC





Chapitre VI

Partie bonus

Si vous avez correctement réalisé le cahier des charges très précis de la partie obligatoire, voici les bonus que nous vous proposons de réaliser :

- Apporter une touche de réalisme à l'ensemble
- Tout autre bonus que vous jugerez utile, si vous pensez que vos correcteurs partageront votre définition de l'utilité

Chapitre VII

Rendu et peer-évaluation

Rendez-votre travail sur votre dépôt GiT comme d'habitude. Seul le travail présent sur votre dépôt sera évalué en soutenance.