Rapport du projet CRV

Cindy Even (c9even@enib.fr)

Guillaume Biannic (g9bianni@enib.fr)

Décembre 2014

Sommaire

Enoncé du projet :

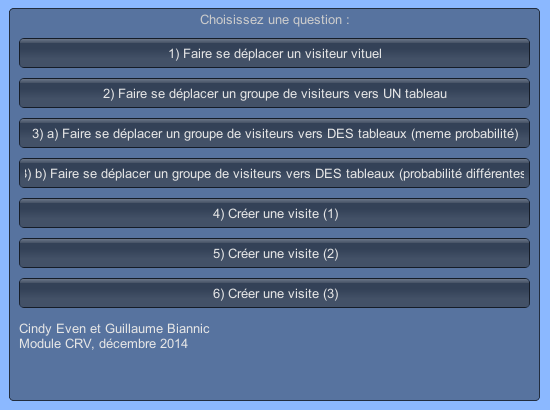
On considère un musé contenant une quarantaine de tableaux (ou posters, vieilles cartes postales, …). Il s’agit de guider l’utilisateur dans ce musée en tenant compte de ses préférences. Plutôt que d’utiliser un système de fléchage (dynamique ou non) ou d’incarner un système de guidage par un acteur virtuel, on préfère ici utiliser les autres visiteurs (représentés par des acteurs virtuels) pour inciter l’utilisateur à se diriger vers un tableau plutôt qu’un autre.

L’hypothèse qui est faite ici est la suivante :

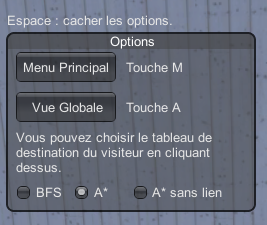
*Un attroupement de visiteurs devant un tableau doit inciter l’utilisateur à aller vers ce tableau.*

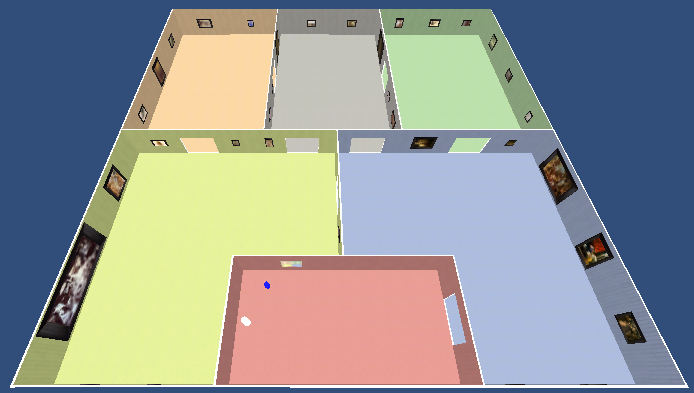
Pour réaliser ce projet nous avons décidé d’utiliser Unity3D.

Menu principal, chaque bouton montre l’implémentation de la réponse à la question associée.



Présenter les options ici aussi (du moins dire que espace ca affiche les options, A ca change la vue, M ca affiche le menu principal, et qu’il y a d’autres options dispo en fonction de la question)





1. Présentation d’Unity3D

Unity3D est un système multiplateforme de création de jeu vidéo développé par Unity Technologies. Ce système comprend un moteur de jeu et environnement de développement intégré (IDE) et utilise entre autre le moteur physique PhysX de Nvidia. Afin de faciliter la compréhension du travail que nous avons réalisé, voici un petit glossaire comprenant différents termes importants utilisés par Unity3D :

- GameObjet (GO) : ce sont les entités les plus importantes dans Unity3D. Tout objet dans l’application est un GO, cependant, ils ont besoin de composants (qui héritent de la classe Component) pour se différencier les uns des autres. On peut voir un GO comme une boîte vide à laquelle il faut ajouter des composants pour en faire un personnage, une lumière, ou des effets spéciaux par exemple.

- Component : les composants (ou Components) sont les pièces fonctionnelles d’un GO. Certains existent déjà dans Unity3D, mais il est possible de créer nos propres composants à l’aide des Scripts.

Voici la liste des composants utilisés dans notre application :

- Transform : c’est le seul composant que tout GO possède. Il permet de définir la position, la rotation et l’échelle du GO dans la scène. Ce composant permet aussi le Parenting, ce concept permet d’attacher un GO à un autre GO, le premier étant alors enfant du second. Un enfant héritera alors du mouvement et de la rotation de son parent.

- Camera : il permet de capturer et d’afficher le monde pour l’utilisateur. Il peut y avoir plusieurs caméras dans la scène mais une seule peut être active à la fois. Pour la visite du musée nous avons deux points de vue (et donc deux caméras). L’un est le point de vue du visiteur (First Person View) et le second permet de voir le musé en entier. Pour le FPV il suffit d’attacher la caméra au visiteur et de la placer au niveau des yeux.

- Scripts : comme dit précédemment, Unity nous permet de créer nos propres composants à l'aide de scripts. Pour cela il suffit de créer une nouvelle classe héritant de MonoBehaviour. Nous pourrons alors utiliser les méthodes Start(), Update() et autres pour implémenter les fonctionnalités voulues.

- Start() est appelée à la toute première frame.

- Update() est appelée toutes les frames.

- OnGUI() est appelée pour le rendu et la gestion des événements de l'interface graphique.

Nous pouvons aussi créer des Scripts contenant des classes n’héritant pas de MonoBehaviour et ne seront alors pas considérés comme des composants mais comme des classes normales.

- Rigidbody : c’est le composant principal qui permet à un objet d’être soumis aux lois de la Physique. En ajoutant ce composant à un objet, ce dernier sera immédiatement répondre à la gravité. Il nous permet aussi d’appliquer des forces sur l’objet.

- Collider : ce composant permet de définir la forme utilisée pour les collisions. Plusieurs formes sont possibles comme un cube, une sphère, un cylindre, ou même la forme même de l’objet : le mesh. Plus la forme est simple et plus les calculs de collisions seront rapides.

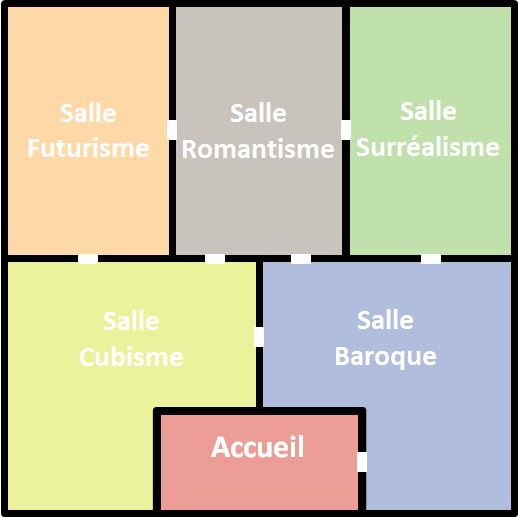
Il est aussi possible d’utiliser le moteur physique pour détecter quand un Collider pénètre dans un autre Collider sans créer de collision. Pour cela il suffit d’utiliser la propriété Trigger du Collider. L’objet ne se comporte alors pas comme un objet solide et permettra simplement d'autres Colliders de passer à travers. Quand un Collider entre son espace, ceci va appeler la fonction OnTriggerEnter().

- Tag : Un tag est un mot qui permet d’accéder à un ou plusieurs GO. Nous pouvons les utiliser dans les Scripts pour trouver un GO qui contient le Tag souhaitée. Ce résultat est obtenu en utilisant la fonction GameObject.FindWithTag().

1. Création du musée :
   1. Objets 3D constituant la scene :

Musé construit sous 3dsMax

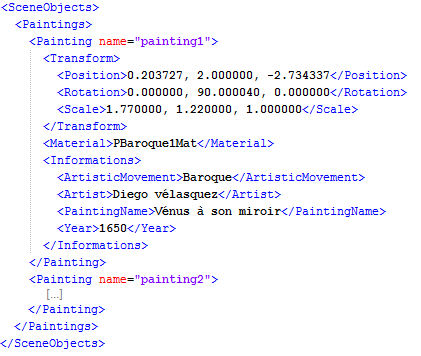
Différentes salles selon le mouvement artistique. Les tableaux sont répartis de façon uniforme dans les salles.



Tableaux aussi fait avec 3dsMax, ils ont bien une face avant et arrière différente (ajouter image)



Fichier XML et classes pour le positionnement des tableaux de façon auto.

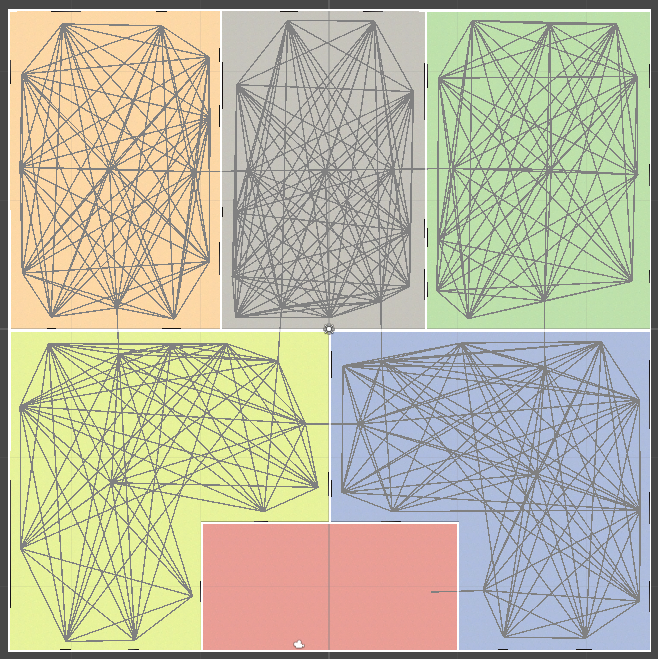


* 1. Déplacements de l’utilisateur

Déplacement de l’utilisateur se fait grâce aux touches clavier (flèches ou touches ZQSD au choix). La caméra bouge de façon à suive la sourie sur l’écran.

Quand l’utilisateur est proche d’un tableau et que le curseur de la sourie est au-dessus du tableau, des informations sur le tableau s’affiches.

* 1. Graphe de navigation



Pseudo code ou diagramme UML de la classe Point et dire que on a fait la même chose que pour les tableaux (XML + classes)

1. Faire se déplacer un visiteur virtuel :

Parcours de graphe

BFS

A\*

A\* sans lien

1. Faire se déplacer un groupe de visiteurs virtuels vers UN tableau :

Steering (Boids)

1. Faire se déplacer un groupe de visiteurs virtuels vers DES tableaux :

Algo de sélection roulette

Pour a) tous les tableaux ont la même fitness

Pour b) la fitness varie en fonction des tableaux.

1. Créer une visite (1)
2. Créer une visite (2)
3. Créer une visite (3)

Bibliographie