



UNIVERSITÉ DE FRANCHE-COMTÉ

RAPPORT DE PROJET TUTEURÉ
LICENCE INFORMATIQUE 3^{ÈME} ANNÉE

Création d'un Third-Person Shooter avec le Shine Engine



Virgil MANRIQUE Quentin GUILLIEN

Encadrant :
Sylvain GROSDÉMOUGE

Année 2015-2016

Remerciements

Ce projet tuteuré n'aurait pu être réalisé sans l'aide de plusieurs personnes que nous tenons à remercier.

Tout d'abord, nous remercions notre encadrant M. Sylvain GROSDÉMOUGE pour nous avoir guidé et prodigué ses conseils durant la réalisation de ce projet.

Ensuite, nous voulons remercier M. Bastien SCHATT, pour nous avoir aidés lors de notre apprentissage de l'utilisation du Shine Engine ainsi que pour sa réactivité lorsque nous avons rencontré des problèmes.

Nous voulons également remercier M. Nicolas DIOT, pour son assistance apportée par rapport au fonctionnement de l'éditeur de niveaux du Shine Engine et des ressources graphiques.

Nous tenons enfin à remercier toutes les personnes qui nous ont aidés de près ou de loin.

Table des matières

1	Cadre du projet	3
1.1	Les Third-Person Shooter	3
1.2	Les moteurs de jeu	4
1.3	Le SHINE ENGINE	4
2	Préparation	5
2.1	Installation des outils	5
2.1.1	Les indispensables	5
2.1.2	Pour plus de confort	5
2.2	Formation au SHINE ENGINE	6
3	Réalisation	7
3.1	hi mom	7
3.2	Problèmes rencontrés	7
3.2.1	Problèmes logiciels	7
3.2.2	Problèmes de code	7
4	Bilan	9

1 Cadre du projet

Dans le cadre de notre 3^{ème} année de licence informatique à l'université de Franche-Comté, il nous a été demandé de réaliser un projet dans le cadre du module projet tuteuré. La durée du projet s'étendait d'Octobre à Mars.

Parmi les sujets proposés, l'un d'entre eux a particulièrement retenu notre attention. Il s'agissait du développement d'un Third-Person Shooter en utilisant un moteur jeu : le SHINE ENGINE développé par M. Sylvain GROSDÉMOUGE.

Nous avons pu nous voir attribuer ce sujet l'ayant placé en première position dans notre liste de choix.

Pour comprendre l'intérêt que nous portions à ce sujet, il faut nous pencher sur les différents éléments qui le constituent, à savoir les Third-Person Shooter, les moteurs de jeu, et le SHINE ENGINE.

1.1 Les Third-Person Shooter

Un TPS (Third-Person Shooter ou jeu de tir à la troisième personne en français) est un sous-genre des jeux de tir et donc des jeux d'action. Les jeux de tir mettent souvent la rapidité et la réactivité du joueur à l'épreuve. L'objectif de ce genre de jeu est de vaincre ses ennemis en utilisant une arme de tir. La particularité des TPS¹ est que le joueur voit son personnage de manière externe contrairement aux FPS² où le joueur voit à travers les yeux de son personnage.



FIGURE 1.1 – Un exemple de TPS : Resident Evil 4

-
1. **Third-Person Shooter**
 2. **First-Person Shooter** : Jeu de tir à la 1^{ère} personne

1.2 Les moteurs de jeu

“Un moteur de jeu est un ensemble de composants logiciels qui effectuent des calculs de géométrie et de physique utilisés dans les jeux vidéo. L’ensemble forme un simulateur en temps réel souple qui reproduit les caractéristiques des mondes imaginaires dans lesquels se déroulent les jeux. Le but visé par un moteur de jeu est de permettre à une équipe de développement de se concentrer sur le contenu et le déroulement du jeu plutôt que la résolution de problèmes informatiques.”

– *Définition Wikipédia*

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

1.3 Le Shine Engine

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

2 Préparation

2.1 Installation des outils

Avant de commencer à coder, il a d'abord fallu installer plusieurs outils logiciels.

Tous les outils mentionnés ont été utilisés, dans notre cas, sur Windows uniquement.

2.1.1 Les indispensables

Les trois logiciels indispensables sont le **Shine SDK**¹, le **Shine Editor**² et **Microsoft Visual Studio 2010**.

Le **Shine Editor** et le **SDK** ne requérant pas d'installation, n'ont pas posé de problèmes dans un premier temps. En revanche, il n'a pas été facile de retrouver la version 2010 de **Visual Studio**, car aujourd'hui, Microsoft ne propose que la version *Community* de leur logiciel. **Visual Studio** est un **IDE**³. Il en existe d'autres, mais le **Shine Engine** a été développé et prévu pour être intégré à **Visual Studio**.

Même après avoir installé les outils **Shine**, il ne peuvent pas encore être exécutés indépendamment. En effet, ils requièrent le **DirectX SDK** mais aussi **Microsoft .NET Framework** (version 4.0 ou supérieur).

2.1.2 Pour plus de confort

Bien qu'ils ne soient pas réellement indispensables, les outils suivants nous ont été extrêmement utiles.

Nous avons utilisé **Git**, plus précisément **Github**, l'application graphique de **Git** pour Windows. **GitHub** est un service web d'hébergement et de gestion de développement de logiciels utilisant **Git**, le logiciel de gestion de versions décentralisé.

Nous avons également utilisé **Trello**, un outil de gestion de projet en ligne permettant d'assigner facilement des tâches à des utilisateurs.

1. SDK = Software Development Kit (trousse de développement logiciel en Français)

2. Editeur graphique de **Shine** permettant d'ajouter facilement des éléments visuels dans un jeu

3. IDE = Integrated Development Environment (Environnement de Développement en Français)

GIMP⁴ est un outil d'édition et de retouche d'image. GIMP a été utile pour créer des **sprites**⁵ simples.

En complément de GIMP nous avons utilisé **XnView**, qui nous a permis de vérifier le formats de certains fichiers, et de convertir au bon format si besoin.

Enfin, nous avons utilisé **L^AT_EX** pour la rédaction de ce rapport.

2.2 Formation au Shine Engine

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

4. GNU Image Manipulation Program

5. Élément graphique qui peut se déplacer sur l'écran. Dans notre cas, cet élément était une simple image en deux dimensions

3 Réalisation

3.1 hi mom

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

3.2 Problèmes rencontrés

3.2.1 Problèmes logiciels

Nous l'avons mentionné dans la partie Installation (page 5), trouver la version complète de **Microsoft Visual Studio 2010** n'a pas été tâche facile.

Cependant, cela n'a pas été le plus important des problèmes logiciel que nous avons eu. En effet, il nous est arrivé par exemple de ne plus pouvoir lancer l'éditeur de **Shine** alors qu'il marchait auparavant, sans savoir pourquoi. Réinstaller l'éditeur, ou installer une version plus récente résolvait en général les problèmes rencontrés. Il nous est également arrivé de simplement ne pas du tout pouvoir faire fonctionner le **Shine Editor** sur un PC (la seule option était ici d'utiliser un autre PC).

3.2.2 Problèmes de code

Le problème majeur qui nous a coûté le plus de temps est celui-ci :

```
1  /// @todo comment
```

Cette ligne, combinée à l'inexistence de documentation, nous a souvent laissé dans le flou quand à la manière de procéder pour effectuer des tâches parfois assez simples. La seule chose dont on pouvait s'aider était les fichiers d'entête dans les dossier du **Shine SDK** : en lisant les définitions des méthodes et les noms des paramètres, on pouvait au mieux *deviner* la façon d'utiliser ces méthodes.

Bien sûr, quand on ne pouvait plus avancer, nous avons pris contact avec l'équipe de **Shine**. Cependant, la réponse n'étant évidemment pas immédiate, il nous est arrivé de rester bloquer des après-midis entiers à essayer de faire marcher un bout de code.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

4 Bilan

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.