



# MOONLIGHT IRA

Anthony GRANVILLE, Camille FORESTIER,  
José RODRIGUEZ, Paul ROBAIN

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Notre projet</b>	<b>4</b>
2.1	Origine du projet . . . . .	4
2.2	Nature du projet . . . . .	4
<b>3</b>	<b>Objet de l'étude</b>	<b>5</b>
3.1	Objectif du Projet . . . . .	5
3.2	Apports Personnels . . . . .	5
<b>4</b>	<b>État de l'art</b>	<b>6</b>
4.1	Dans le même genre . . . . .	6
4.2	Inspirations en terme de Game Design . . . . .	9
<b>5</b>	<b>Découpage du projet</b>	<b>12</b>
5.1	Le choix du moteur de jeu . . . . .	12
5.2	Outils tiers . . . . .	13
5.3	Coûts d'opération . . . . .	13
5.4	Répartition des tâches . . . . .	14
5.5	Planning d'avancement . . . . .	15
<b>6</b>	<b>Conclusion</b>	<b>15</b>



# 1 Introduction

Ce document a pour but de fixer les méthodes et objectifs de notre projet, mais également de montrer nos inspirations, nos progrès, ainsi que la répartition des tâches liées au développement de notre jeu, intitulé *Moonlight Ira*. Notre groupe, la *Team Lotus*, est composé de quatre élèves en première année à l'*EPITA* : Paul ROBAIN, José RODRIGUEZ, Anthony GRANVILLE et Camille FORESTIER.

*Moonlight Ira* affichera des graphismes de style rétro, mais basés sur un univers futuriste, puisqu'il se déroulera sur une autre planète que la Terre. *Moonlight Ira* sera une expérience co-op, dans laquelle les joueurs pourront faire comme bon leur semble dans une ville spatiale. Chacun de ses recoins donnera l'occasion de partir à l'aventure pour accomplir des objectifs aussi variés que loufoques. Elle sera remplie d'habitants au comportement intelligent, et capables d'interagir avec le(s) joueur(s). Fun, Drame et Rebondissements seront au rendez-vous !

Ce jeu sera réalisé avec le *Godot Game Engine*, un moteur de jeu multiplateforme supportant le C#, à l'instar d'*Unity*.

Ce projet nous permettra de nous développer, que ce soit en tant qu'individu ou en tant que groupe, en faisant appel à nos capacités personnelles telles que le codage, la création d'objets, l'animation, la résolution de problèmes, la persévérance, mais aussi et surtout, l'entraide et la coopération.

## 2 Notre projet

### 2.1 Origine du projet

Aimant les jeux *open-world*<sup>1</sup> et les jeux multijoueur, il nous semblait évident de créer un monde que le joueur pourrait explorer librement. Afin d'indiquer visuellement la progression du joueur, le monde changera peu à peu au fil de sa quête. Si le joueur rencontre des difficultés, il pourra faire appel à un ami pour l'aider<sup>2</sup>. Cependant, pour contrebalancer cette aide, la difficulté du jeu pourra augmenter, requérant une certaine coordination entre les joueurs pour avancer.

Cet univers est inspiré d'œuvres nous ayant touchés, tant par leurs cinématographies que par leurs messages. *Wall-E*<sup>3</sup> et *Interstellar*<sup>4</sup> en sont les meilleurs exemples, c'est pourquoi notre esthétique en sera grandement inspirée.

### 2.2 Nature du projet

*Moonlight Ira* est un jeu vidéo d'action-aventure se jouant en vue de 3/4, en 2D. Dans ce sandbox, le joueur sera libre de parcourir la carte comme bon lui semble, et d'effectuer les missions/niveaux dans l'ordre souhaité. Celui-ci incarne un astronaute coincé sur une planète inconnue dont il cherche à repartir au plus vite. Il progressera dans son périple en explorant, de fond en comble, une ville futuriste, et en accomplissant des missions. Il pourra ainsi obtenir de nouvelles armes et de nouvelles capacités. Le joueur devra faire preuve d'inventivité et de réactivité pour les utiliser au mieux, et espérer échapper à ce monde étrange.

---

1. en monde ouvert

2. si tant est qu'il ait des amis

3. réalisé par Andrew Stanton en 2008

4. réalisé par Christopher Nolan en 2014

## 3 Objet de l'étude

### 3.1 Objectif du Projet

Ce projet va nous permettre d'utiliser nos connaissances acquises au cours de l'année afin de mener à bien notre jeu vidéo. Il sera donc nécessaire de concevoir l'univers dans lequel le joueur évolue, que ce soit dans son gameplay, son audio ou son design par exemple. Il doit être original, tout en se rattachant à des éléments connus du public. Le but est donc d'être innovant en réinventant certaines mécaniques de jeu "classiques" tout en respectant les délais imposés.

### 3.2 Apports Personnels

Ce projet préparera notre groupe au monde professionnel et à ses défis, en nous entraînant à :

- suivre un planning d'avancement ;
- développer la cohésion et l'entraide au sein de notre groupe ;
- partager nos connaissances les uns avec les autres ;
- respecter les avis de chacun.

De manière plus personnelle, ce projet permettra également :

- de développer nos compétences en *POO*<sup>5</sup> en *C#* ;
- de découvrir le codage d'une Intelligence Artificielle ;
- d'étudier la conception d'un site Web<sup>6</sup> ;
- de perfectionner la création de sprites<sup>7</sup> et leurs animations grâce au logiciel *Aseprite* ;
- de nous initier à l'enregistrement d'effets sonores et à l'écriture de musiques correspondant à l'univers du jeu.

---

5. Programmation Orientée Objet

6. en utilisant du PHP, HTML, CSS et JavaScript

7. Dessins de type "pixel-art"

## 4 État de l'art

### 4.1 Dans le même genre

#### Grand Theft Auto

L'un des premiers jeux du genre serait *Grand Theft Auto*, sorti en 1997 sur MS-DOS et MacOS, un jeu d'action-aventure en vue du dessus. Malgré ses grandes différences avec les opus suivants, le jeu est un succès commercial, acclamé pour sa "grande rejouabilité" et la "liberté" importante dont dispose le joueur, à l'époque sans précédent.



FIGURE 1 – Grand Theft Auto, sorti en 1997

Cependant, on note que les graphismes, déjà dépassés à l'époque, ont relativement mal vieilli.

Nous tâcherons donc d'en reprendre la rejouabilité et le degré de liberté laissé au joueur, sans pour autant négliger les graphismes et le World Building.

## Retro City Rampage

Un exemple plus récent, et plus réussi du genre, serait *Retro City Rampage*, sorti en 2012. Le jeu s'inspire directement du premier GTA, reprenant son concept primaire, avec cette fois une vue de 3/4, et non pas directement une vue du dessus.



On note les collines avec des yeux et le tuyau vert, références aux jeux Mario.

Cependant, il se démarque par son style rétro. Utilisant des graphismes et sons 8-bit, il comporte des références, autant pop que vidéoludiques des années 80-90. Il s'agit d'une véritable lettre d'amour au rétro.

De plus, les diverses missions permettent de varier le gameplay, voire même le style de jeu lui-même, rendant le jeu encore plus rejouable que son prédécesseur. L'élément clé de ce jeu est ainsi l'avalanche de contenu "simple", mais **varié**. Grâce à cela, le jeu devient un succès critique et commercial, s'écoulant à plus de 400 000 exemplaires en 2014. Pas mal pour un jeu indé ! Anecdote : pour faire écho à son aspect rétro, le jeu à été porté à l'identique sur DOS.

## Saints Row IV

Cependant, depuis *Grand Theft Auto*, le genre a bien évolué. Étant donné que ce type de jeu tend à servir de défouloir, presque tous les jeux du genre (*Retro City Rampage* exclus) ont pris le chemin du réalisme et de la caricature.

Saints Row IV ne fait pas exception à la règle. Sorti le 23 août 2013, il pousse tous les éléments du genre au maximum. Le joueur incarne un président des États-Unis déchu, combattant des extraterrestres avec des super-pouvoirs dans une simulation informatique. Le ton a beau être excessif, il faut reconnaître de très bonnes idées en terme de gameplay, notamment les pouvoirs qui ajoutent un degré de liberté au joueur, lui donnant une sensation de puissance sans précédent.



FIGURE 2 – Un exemple de gameplay de Saints Row IV

Cependant, la direction prise par les jeux du genre, poussant à l'hyper-réalisme et au toujours plus caricatural nous paraît ridicule. Voilà pourquoi nous souhaitons retourner aux racines du genre, en réalisant un jeu en 2D.



## 4.2 Inspirations en terme de Game Design

### Minishoot'

En terme de Game Design et d'expérience Utilisateur cette fois, *Minishoot'*, alors qu'il est encore en développement, pourrait être un cas d'école.



FIGURE 3 – Minishoot'

Le jeu est un twin-stick shooter dans lequel on explore un monde étrange, peuplé de petits vaisseaux. Le principal point d'intérêt pour nous ici est son interface. En effet, cette dernière reste minimaliste par choix, n'affichant les informations que lorsqu'elles sont réellement nécessaires.

On note également que les ennemis deviennent de plus en plus rouge lorsqu'ils subissent des dégâts, remplaçant ainsi une barre de vie, et laissant plus de place au joueur pour admirer les graphismes et animations.

On tâchera donc de garder cette philosophie en tête en concevant notre jeu, mais nous allons plus particulièrement nous inspirer de l'aspect graphique et design de ce jeu-là. Ce dernier est en partie ce que nous cherchons à faire pour notre jeu : lui donner un aspect coloré, relativement rond, fluide, interactif, et simple d'utilisation. Autre point d'intérêt : ce jeu est en monde ouvert, ce qui permet au joueur de parcourir la carte et d'avancer dans l'ordre souhaité.

## The Legend of Zelda : A Link Between Worlds

Zelda ALBW est un excellent exemple de World Design.



FIGURE 4 – La carte telle que vue en jeu

Le(s) monde(s) (il y en a deux!) sont divisés en 7 zones chacun. Chacune de ces zones est l'occasion pour le joueur de découvrir un environnement différent, et d'apprendre à maîtriser de nouvelles capacités. Ces dernières ayant été conçues **après** le personnage, elles forment un terrain de jeu sur-mesure, permettant au joueur de tirer le meilleur de son personnage.

## Kynseed



FIGURE 5 – Un exemple de bâtiment dans Kynseed

Ici, le joueur peut ouvrir une boutique, cultiver la terre, se lier avec les différents personnages, récupérer des matériaux pour faire des armes, cuisiner ou préparer des remèdes, et, le plus important, explorer le vaste pays de Quill. Il peut aussi combattre des ennemis, ou même explorer des donjons.

Les points que nous avons retenus sont les suivants. En premier lieu, le degré de liberté important laissé au joueur lui permet de se balader comme bon lui semble tout en ayant un objectif en tête. Il peut faire chaque région dans l'ordre qu'il souhaite. Ensuite, les villes et les boutiques sont animées. Dans *Kynseed*, les PNJs<sup>8</sup> rendent les villes vivantes, et donnent l'impression d'un monde cohérent.

---

8. Personnages Non Joueurs

## 5 Découpage du projet

### 5.1 Le choix du moteur de jeu

Afin de mener à bien ce projet, plusieurs outils seront utilisés. Le premier, et sans doute le plus important, sera le moteur de jeu Godot.



FIGURE 6 – Le logo du moteur de jeu Godot Game Engine

Notre choix s'est en effet porté sur ce dernier, plutôt que sur le plus traditionnellement utilisé Unity, et ce pour plusieurs raisons.

La première est sa légèreté. En tant que moteur de jeu, que ce soit pour le concepteur ou pour le joueur, Godot est bien plus léger qu'Unity. En résultent des jeux plus légers, et surtout, plus performants. Là où l'espace occupé par une installation d'Unity se chiffre en Go, une installation de Godot n'occupe qu'une centaine de Mo.

La deuxième est sa conception. Bien que les deux moteurs aient en commun le support du C#, des capacités 2D et 3D, et un support communautaire important, leurs paradigmes diffèrent. Alors que par sa conception Unity pousse à créer des objets "tout-puissants"<sup>9</sup>, moins optimisés, Godot encourage l'interaction entre objets pour éviter toute redondance.

---

9. Comprendre : "entièrement autosuffisants"

Enfin, Godot dispose de puissants outils intégrés (notamment concernant l’animation d’objets) qui profiteront grandement à notre projet, et qui ne sauraient trouver d’égal dans les éléments proposés par Unity.

## 5.2 Outils tiers

Pour réaliser toutes les ressources liées à notre projet, qu’il s’agisse d’images, de sons, de musiques, ou autres, nous aurons recours à quelques logiciels tiers :

- Aseprite, pour réaliser et animer des sprites<sup>10</sup> ;
- Audacity, pour capturer et éditer des sons ;
- ChipTone, pour réaliser des effets sonores rétro<sup>11</sup> ;
- Bosca Ceoil, pour réaliser des musiques rétro.

## 5.3 Coûts d’opération

Les outils utilisés ont été soigneusement choisis pour leur efficacité et leur faible coût (ils sont open-source et/ou gratuits). Ainsi, les coûts d’opération se limiteront aux coûts liés au nom de domaine du site Web, ce dernier étant hébergé sur un service gratuit comme *Github Pages*.

La publication du jeu se fera à travers *itch.io*, sans coût associé.

---

10. images de type "pixel-art"

11. type "8-bit" ou "16-bit"

## 5.4 Répartition des tâches

Les tâches seront réparties de la manière suivante :

Conception	José	Paul	Anthony	Camille
Design				
Game design			R	S
Character design			S	R
World design			S	R
UI/UX design		R		S
Programmation				
General programming		S	R	
AI Programming	R		S	
Multiplayer-related	S		R	
Website		S		R
Son				
Effets Sonores (SFX)	R	S		
Musique	S	R		
Sprites				
Personnages			R	S
Bâtiments	R	S		
Véhicules	S	R		
Objets	R			S

## 5.5 Planning d'avancement

Section	Période 1	Période 2	Période 3
Gameplay	30%	70%	100%
Réseau	10%	50%	100%
IA	20%	60%	100%
Visuels	10%	55%	100%
Audio	10%	60%	100%
Site Web	40%	100%	100%

## 6 Conclusion

Bien qu'il nous demande une certaine coordination, un important travail d'équipe et un investissement conséquent de chacun d'entre nous, nous sommes déjà impatients de commencer ce projet. Une activité de la sorte nous donnera un avant-goût des tâches d'un ingénieur, que ce soit par les différents aspects techniques à gérer, ou par les délais qui nous sont imposés. Sa simple planification nous en apprend déjà beaucoup sur nous-mêmes et sur nos objectifs, bien plus que n'importe quelle théorie aurait pu le faire.

Le projet dans lequel nous nous engageons pour les sept prochains mois est, bien plus qu'un projet scolaire, une grande aventure collective ! Nous ferons de notre mieux pour atteindre nos objectifs, qui sait jusqu'où ils pourraient nous mener...