



10 AVRIL 2020

RAPPORT DE PROJET

UE PROJET : JEUX 2D

LAURENT QUENEL, FLORIAN PIAULT

INFO GROUPE 1

Frederic Fürst, Yu Li

Table des matières

I.	INTRODUCTION	3
II.	LE CONTEXTE	3
1.	<i>Description du projet</i>	3
2.	<i>Objectif du projet</i>	3
3.	<i>Description générale du besoin</i>	3
III.	DESCRIPTION FONCTIONNELLE	4
1.	<i>Menu principal</i>	4
2.	<i>Mettre en pause</i>	4
3.	<i>Quitter le jeu</i>	4
4.	<i>Se déplacer</i>	4
5.	<i>Tirer</i>	5
6.	<i>Monnaie en jeu</i>	5
7.	<i>Gagner ou perdre de la vie</i>	5
8.	<i>Gagner le jeu</i>	5
IV.	ETUDE DE L'EXISTANT	5
V.	CONTRAINTE TECHNIQUE.....	6
VI.	STRUCTURE ET CONTROLES.....	6
1.	<i>Maquettage</i>	6
2.	<i>Contrôles</i>	7
VII.	LA CHARTE GRAPHIQUE	8
1.	<i>Couleurs et thème</i>	9
2.	<i>Le logo</i>	9
3.	<i>Typographie</i>	9
VIII.	TECHNOLOGIE.....	10
IX.	ETUDE D'EVOLUTION	10
1.	<i>Mouvements supplémentaires pour le personnage</i>	10
2.	<i>Ennemis variés</i>	10
3.	<i>Plus de carte de jeu</i>	10
4.	<i>Manette</i>	11
5.	<i>Langue du jeu</i>	11
6.	<i>Optimisations</i>	11
X.	PLANNIFICATION	11
1.	<i>Liste de tâches</i>	12
2.	<i>Diagramme de PERTT</i>	12
XI.	ANALYSE	13
1.	<i>Faits coïncidents avec le cahier des charges</i>	13
2.	<i>Faits en désaccord avec le cahier des charges</i>	13

XII. CONCEPTION	13
1. <i>Sujet du projet.....</i>	13
2. <i>Langage.....</i>	14
3. <i>Direction artistique.....</i>	14
4. <i>Contrôles</i>	14
5. <i>Environnement de travail</i>	14
XIII. CODAGE.....	14
1. <i>Environnement.....</i>	15
2. <i>Structures utilisées</i>	15
XIV. TEST.....	15
1. <i>Le joueur gagne la partie.....</i>	15
2. <i>Le joueur perd la partie</i>	17
3. <i>Démonstration</i>	19
XV. BILAN.....	19
1. <i>Difficultés</i>	19
2. <i>Améliorations</i>	19
3. <i>Bénéfices</i>	19
XVI. CONCLUSION	21
XVII. BIBLIOGRAPHIE.....	22

I. INTRODUCTION

Dans le cadre de l'Unité d'Enseignement « Projet » nous avons pour but de mener à bien un projet dans son intégralité. Celui-ci sera réalisé en binôme sur une période d'environ 8 semaines. La gestion de ce projet nous permettra de nous familiariser à la mise en place de projet, ce qui peut être fréquent dans le milieu professionnel.

Le projet n'étant pas directement imposé, nous sommes amenés à faire un choix parmi différents thèmes présentés. Nous avons donc choisi de réaliser un jeu en 2D orienté *shoot'em up* (c'est-à-dire un jeu où le joueur doit tuer un certain nombre d'ennemis dans le but de progresser dans celui-ci.). La suite de ce document nous détaillerons les diverses étapes de conception du projet mais aussi la phase de test et les conclusions que nous pourrons en tirer.

II. LE CONTEXTE

1. Description du projet

Comme nous l'avons dit précédemment, nous avons pris la décision de réaliser un jeu. Suite à un *brainstroming* nous avons pris diverses directives quant à ses spécifications. Nous avons décidé que le jeu serait un jeu 2D mais avec une vue du dessus. Le joueur incarnera un personnage dans un monde alternatif où celui-ci devra survivre dans le but de progresser. Ce personnage sera en possession de diverses armes qu'il récupérera tout au long de son aventure. Celles-ci lui permettront de venir à bout d'ennemis de plus en plus fort, voire même venir à bout du jeu lui-même. Le jeu se situera dans une université où le joueur devra venir à bout de professeurs infecté par une maladie inconnue.

2. Objectif du projet

Le jeu devra être simple d'utilisation de par ses contrôles mais aussi son interface intuitive. Le joueur se déplacera grâce aux touches directionnelles et pourra tirer sur les ennemis avec un seul clic de souris. Cette simplicité d'utilisation nous permettra de cibler un public plus large. Donc que le joueur soit adepte ou non de jeux vidéo.

Le projet pourra être modulable, c'est-à-dire qu'à l'avenir on peut imaginer ajouter des armes ou des cartes au jeu actuel. On s'assurera aussi que cette extensibilité soit facile dans le but d'étendre facilement les fonctionnalités du jeu.

3. Description générale du besoin

Le jeu donnera la possibilité au joueur de se déplacer dans diverses cartes et salles. Lors de son aventure il rencontrera divers ennemis dont il pourra venir à bout grâce aux armes en sa possession. Cependant on peut imaginer des salles sans ennemis où le joueur devra simplement trouver des objets cachés ou non. Lorsque le joueur vaincra un ennemi, il aura la possibilité de récupérer différents objets comme de la vie ou de la monnaie. Cette monnaie

pourra être dépensée dans un magasin. Le but du jeu étant de vaincre le chef des ennemis de chaque carte.

III. DESCRIPTION FONCTIONNELLE

Dans cette partie, pour bien comprendre les fonctionnalités du projet, nous segmenterons ces dernières en différents modules.

1. Menu principal

Pour jouer au jeu il est nécessaire de pouvoir commencer une nouvelle partie. Cette nouvelle partie sera accessible sur le menu d'accueil du jeu. On y aura accès grâce à un clic de la souris sur le bouton correspondant. Une fois ceci fait l'utilisateur se verra lancer dans une partie et pourra jouer librement. (Cf. [Structure de l'application](#))

De la même manière le joueur pourra quitter le jeu directement grâce à un bouton ou tout simplement en appuyant sur la croix de la fenêtre.

2. Mettre en pause

Au cours d'une partie le joueur pourra mettre, quand bon lui semble, la partie en pause. Cette pause pourra être accessible grâce à la touche « échap ». L'utilisateur aura accès à un menu de pause. Le menu permettra de suspendre l'activité du joueur pendant le temps qu'il voudra. Celui-ci pourra donc relancer quand il le souhaitera le jeux grâce à un bouton. (Cf. [Structure de l'application](#))

3. Quitter le jeu

Comme nous l'avons dit précédemment le joueur pourra quitter le jeu grâce à des boutons dédiés. Nous aurons un bouton sur le menu pause mais aussi un sur le menu principal. Aussi l'utilisateur pourra tout simplement fermer la fenêtre pour mettre fin au jeu.

4. Se déplacer

Comme le but du jeu étant de venir à bout des différents ennemis, il est indispensable pour le joueur de pouvoir se déplacer dans le monde. Ainsi le joueur pourra librement se déplacer dans les zones accessibles et ce, grâce aux touches directionnelles du clavier. Le personnage que le joueur incarnera pourra donc se déplacer suivant deux axes (vertical, horizontal). Nous ne gèrerons pas de troisième axe donc le personnage n'aura pas la possibilité de sauter.

5. Tirer

Le joueur aura accès à une arme dès le début du jeu et donc pourra l'utiliser à sa guise. L'arme tirera un projectile en ligne droite dans la direction de la souris. Ce projectile aura pour but d'éliminer les différents ennemis auxquels le joueur fera face. Ce tir sera accessible grâce au bouton gauche de la souris.

6. Monnaie en jeu

Lorsque le joueur éliminera des ennemis ceux-ci pourront lui donner des récompenses comme de l'argent ou bien de la vie (Cf. [Gagner ou perdre de la vie](#)), les ennemis les plus difficiles rapportant plus d'argent. Cette monnaie permettra au joueur d'acheter des améliorations dans des boutiques dédiées. On peut imaginer que celui-ci achètera une vie supplémentaire pour tout le reste de son aventure.

7. Gagner ou perdre de la vie

La durée de vie du personnage incarné n'est pas éternelle. On mettra donc en place un système de vie. Lorsque le joueur se fera toucher par un ennemi celui perdra de la vie. Si le joueur n'a plus de vie alors il a perdu. On lui proposera de rejouer (reprendre au début) ou alors de quitter le jeu. Cependant on pourra ramasser de la vie sur certains ennemis.

8. Gagner le jeu

Le jeu n'étant pas infini, celui-ci possède une fin. Pour finir le jeu il vous faudra éliminer un certain ennemi, le plus fort d'entre tous. Une fois éliminé, le joueur aura accès, de la même manière que pour la défaite, à un écran lui permettant de rejouer ou quitter le jeu.

IV. ETUDE DE L'EXISTANT

Le jeu que nous voulons créer est un shoot them up. Il en existe une multitude sur le marché. Nous pouvons citer par exemple *Enter the Gungeon*, *The Binding of Isaac* ou bien *Hotline Miami*.

Hotline Miami est un jeu sorti initialement en 2012 et jouable sur PC et console. C'est un jeu avec une histoire tordue et un gameplay⁽¹⁾ très rapide et nerveux. Le personnage se déplace de pièce en pièce, éliminant le plus d'ennemi possible pour enfin passer à la pièce suivante.

Comme dit ci-dessus, c'est un jeu rapide : lors d'une perte de partie, le joueur a la possibilité de recommencer à un point de sauvegarde proche. Bien que ce soit un avantage que le jeu soit vif, les possibilités de jeu sont moindres : le personnage n'a accès qu'à une seule arme ce qui peut apparaître lassant au bout de quelque temps de jeu.

Enter the Gungeon, sorti en 2016, et The Binding of Isaac, sorti en 2011, sont tous les deux des jeux dits *roguelike*, c'est-à-dire que chaque partie est différente de la précédente.

Ceci est dû au fait que chaque partie a la possibilité d'avoir un assortiment d'objets différents -générés aléatoirement- et entraîne donc une multitude de synergies différentes entre ses objets.

Les possibilités de jeu sont quasi-infinie sur ces jeux dit *roguelike*. Ils sont aussi difficiles à prendre en main comme la multitude d'ennemi ont tous un paternité de déplacement et d'attaque différents, ce qui peut paraître frustrant au début. Mais ce défaut est aussi la force de ce type de jeu : la frustration ressentie par ces nombreuses erreurs et défaites peut se transformer par une satisfaction.

Ce que nous voulons concevoir et un mélange de ces jeux précédemment présentés. Un *gameplay* vif. Plusieurs objets différents seront présents.

(1) Comment est-ce qu'un jeu est joué

V. CONTRAINTES TECHNIQUES

Quand nous avons imaginé le jeu ; nous nous sommes mis d'accord sur le fait que celui-ci n'autoriserait qu'un seul joueur à jouer. En effet notre jeu sera dépourvu d'un mode multijoueur qu'il soit local ou en réseau.

De même le jeu devra être multiplateforme, celui-ci devra fonctionner sur n'importe quel OS qu'il soit Linux, MacOs ou Windows.

Comme le joueur nécessite de se déplacer et de tirer pour jouer, il est donc obligatoire de posséder un clavier et une souris pour profiter pleinement de l'expérience de jeu proposée.

Aussi comme aucun budget n'est alloué pour la réalisation il sera obligatoire de tout réaliser nous-même, comme les sprites par exemple.

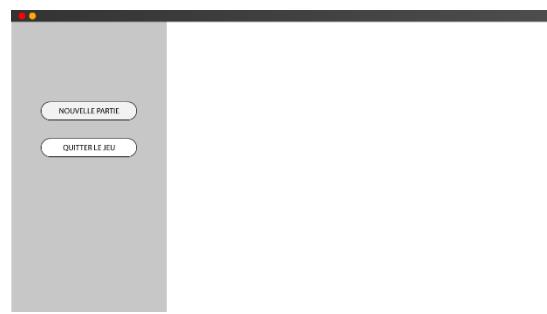
VI. STRUCTURE ET CONTROLES

Cette partie ayant pour but de mieux comprendre le fonctionnement du jeu et ses différents menus, nous la structurerons en deux parties. La première consistant au maquettage du jeu, c'est-à-dire les maquettes des différents menus et interfaces. Et ensuite nous détaillerons les contrôles à la disposition du joueur dans le jeu.

1. Maquettage

a. Menu principal

Le menu principal permettra d'accéder au cœur du jeu. Celui-ci permettra de commencer une nouvelle aventure et de quitter le jeu.



b. Menu pause

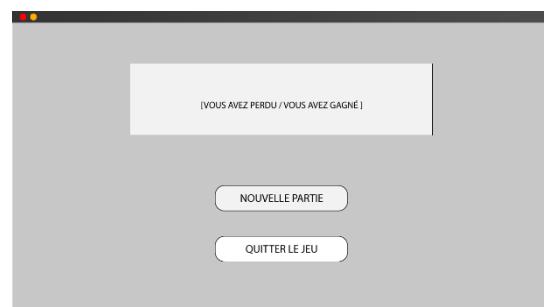
Le joueur aura la possibilité de séquencer ses phases de jeu grâce au menu pause. Il pourra donc mettre en pause sa partie à n'importe quel moment. Sur le menu pause il pourra quitter le jeu ou reprendre le jeu où il en était.



2 Menu Pause

c. Écran de défaite ou victoire

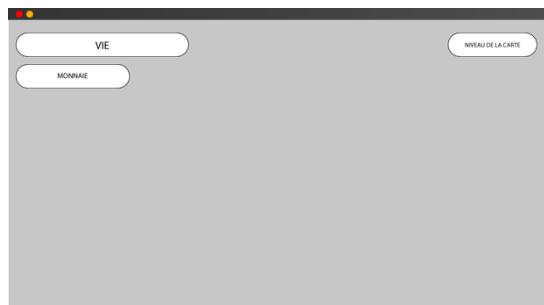
Lorsque le joueur aura perdu ou tout simplement fini le jeu, il sera confronté à cet écran. Celui-ci lui donnera accès à deux boutons permettant de rejouer ou de quitter le jeu.



3 Écran de défaite ou de victoire

d. Interface en jeu

En jeu le joueur aura accès à diverses informations comme sa vie, l'argent qu'il possède sur lui mais aussi sa progression dans le monde. Ces informations permettent au joueur de savoir où il en est dans l'aventure et donc de garder un œil dessus pendant ces phases de jeu.



4 Interface en jeu

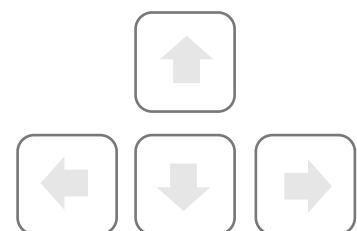
2. Contrôles

Même si les contrôles proposés ne soient pas nombreux il est quand même obligatoire que ceux-ci soient intuitifs et simples d'utilisation. Pour les déterminer nous prenons donc inspiration sur les jeux déjà existants. Ainsi nous aurons les contrôles suivants :

a. Se déplacer dans quatre directions



Comme nous l'avons dit le joueur pourra se déplacer sur deux axes. Pour cela nous utiliserons les touches directionnelles du clavier.



b. Tirer



Comme nous l'avons dit le joueur aura accès à une arme avec laquelle il pourra tirer. Pour cela nous utiliserons le bouton gauche de la souris.



VII. LA CHARTE GRAPHIQUE

Dans le but de trouver une identité au projet, nous avons mis au point une charte graphique. Celle-ci se décompose en plusieurs parties. Dans un premier temps nous parlerons des couleurs et donc du thème, ensuite du logo et nous finirons par la typographie.

1. Couleurs et thème

Pour un souci de cohésion mais aussi pour garder une ambiance cohérente il était obligatoire d'imposer un thème et donc des couleurs à notre jeu. Nous avons donc décidé de situer notre jeu dans un monde retro-futuriste avec des couleurs plutôt prononcées comme le cyan, le magenta et le jaune.



Le jeu s'articulera autour de nuances des couleurs choisies ci-dessus. De cette manière nous aurons un jeu riche en couleur avec une atmosphère.

2. Le logo

Dans le but de donner une identité au jeu nous avons réalisé un logo. Ce logo s'inspire fortement des logos « All Star » utilisés en NBA notamment celui de 2014. On a donc le nom du jeu « UPJV All star : Battle for Jules Verne » décomposé en plusieurs parties. A ce logo nous avons appliqué des nuances des couleurs choisies précédemment soit un ensemble de gradient jaune et magenta, ainsi nous obtenons le résultat suivant :



5 Logo du jeu

Comme le logo comporte un contour blanc, celui-ci est aussi bien applicable sur fond clair que sur fond foncé.

3. Typographie

Pour garder une cohérence avec le thème choisi nous avons dû trouver une police retro-futuriste correspondant à nos attentes. Ainsi nous avons choisi la police suivante :

WAVE ATTACK
ABCDEFGHIJKLMNPQRSTUVWXYZ ABCDEFGHIJKLMNPQRSTUVWXYZ 0123456789

Aussi, il nous fallait une police pour le texte en jeu. Nous avons donc choisi une police pixélisée :

VCR OSD
ABCDEFGHIJKLMNPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789

VIII. TECHNOLOGIE

Pour réaliser un tel jeu, il était important pour nous de travailler dans un environnement familier et à notre portée. C'est pourquoi nous avons choisi le langage C. Comme celui-ci nous est appris lors de notre enseignement, il est donc plus accessible pour nous. Cependant le langage C à proprement parlé ne nous permet pas la création de fenêtre et d'interface graphique comme nous le souhaitons. De ce fait nous utiliserons la bibliothèque SDL2. De cette manière nous pourrons gérer plus facilement les entrées et sorties avec le jeu.

En ce qui concerne le visuel nous réalisons toutes les maquettes et tous les sprites grâce à la suite adobe en l'occurrence Adobe Xd et Adobe Illustrator.

IX. ETUDE D'EVOLUTION

Étant donné la *deadline* imposée nous nous devons d'être réalistes sur les fonctionnalités que le jeu proposera. Cependant nous en avons imaginées beaucoup d'autres, pouvant être implémentées antérieurement à la *deadline* imposée.

1. Mouvements supplémentaires pour le personnage

Les mouvements de notre personnage sont assez simples mais limités, nous pourrons lui ajouter la possibilité de pouvoir sprinter ou faire une roulade par un appui sur une touche quelconque. De cette manière on obtiendrait un jeu plus réaliste.

2. Ennemis variés

Dans le but d'ajouter de la difficulté et du changement dans le jeu, nous aimerons ajouter un plus grand panel d'ennemi. Que ce soient des ennemis avec des apparences ou bien des paternes différents. Il pourrait y avoir des ennemis dits "élites", qui seront des évolutions d'ennemis existants mais avec des statistiques améliorées : plus de point de vie, de dégât, de vitesse d'attaque etc.

3. Plus de carte de jeu

Dans un premier temps, nous créerons une carte qui aura des salles hostiles, un magasin, une salle d'objet et une salle de boss. Mais il serait intéressant de rajouter des niveaux supplémentaires afin de rallonger la durée de vie de notre jeu.

4. Manette

Comme expliqué dans la partie concernant les contrôles du jeu (Cf. [Structure et contrôles](#)), nous avons décidé de contrôler le personnage grâce au clavier ainsi que la souris. Nous proposons donc comme ajout des contrôles faits grâce à une manette. (Xbox 360 / Xbox One)

5. Langue du jeu

Nous proposons notre jeu en français. Mais nous pouvons aussi envisager l'ajout de la langue anglaise. Ce choix serait proposé dans le menu principal. Ainsi le public touché serait plus vaste.

6. Optimisations

Pour peaufiner notre jeu, nous pouvons toujours y apporter des optimisations. Que ce soient la façon dont les boîtes de collisions⁽²⁾ des personnages sont gérées, ou bien comment nous gérons les certaines fonctions tels les changements des sprites⁽³⁾ des personnages etc.

(2) la zone autour d'un personnage ou d'un objet servant à la détection des collisions, aussi appelée masque de collision ou hitbox en anglais.

(3) une image en 2D représentant le personnage

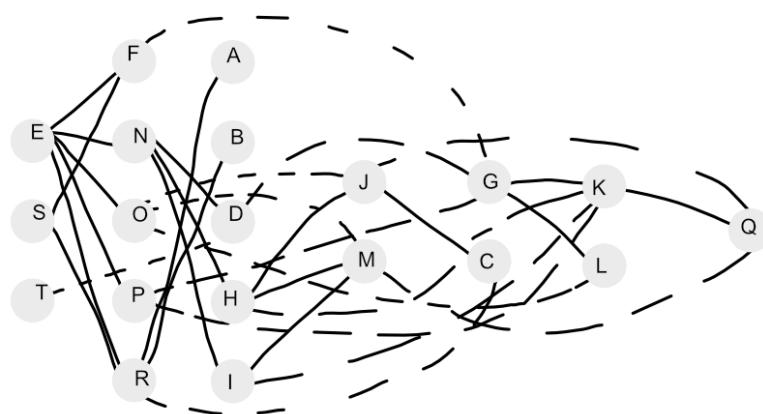
X. PLANNIFICATION

Dans le but de garder un suivi constant sur le projet et d'être organisé dans la création de celui-ci, il nous a été demandé de réaliser un Trello (Cf. [Trello Jeu 2D | UPJV All Star](#)). De la même manière nous avons découpé notre projet en différentes tâches de manière à organiser celles-ci grâce à un diagramme de Pert.

1. Liste de tâches

Tâches	Code	Antériorité	DUREE (en heure)	NIVEAU
REALISATION DU MENU PRINCIPAL	A	R	2	3
REALISATION DU MENU PAUSE	B	R	2	3
REALISATION DU MENU DE VICTOIRE ET DEFAITE	C	J,R	2	5
GESTION DU DEPLACEMENT DU JOUEUR	D	N,T	1	3
CREATION DES SPRITES	E	/	4	1
GESTION DES TIRS	F	S,E	3	2
GESTION DES COLLISIONS	G	P,D,L,F	3	5
GESTION DE LA VIE	H	N	1	3
GESTION DE LA MONNAIE	I	N	1	3
GESTION DES CONDITIONS DE VICTOIRE ET DEFAITE	J	H,O	1	4
GESTION DE LA BOUTIQUE	K	H,I,G,P	3	6
GESTION DES DEPLACEMENTS DES ENNEMIS	L	O,G	3	4
GESTION DES OBJETS LACHES	M	O,I,H	1	4
REALISATION DU PERSONNAGE	N	E	1	2
REALISATION DES ENNEMIS	O	E	1	2
REALISATION DE LA CARTE	P	E	4	2
GESTION DU SON	Q	J,K,M	3	7
IMPLEMENTATION DES BOUTONS	R	E,S	2	2
GESTION DE LA SOURIS	S	/	1	1
GESTION DU CLAVIER	T	/	1	1

2. Diagramme de PERTT



XI. ANALYSE

Dans le but de rédiger un compte rendu cohérent avec le cahier des charges, il est important d'analyser le travail que nous avons fournis durant la phase de conception. Pour cela nous procèderons en deux parties ; la première portant sur tout ce qui coïncide avec ce que nous avons initialement prévu et la dernière portant sur les désaccords avec le cahier des charges.

1. Faits coïncidents avec le cahier des charges

Pour une grande majorité des choses que nous avons prévues dans le cahier des charges, la majorité de ces charges ont été implémentées dans notre jeu. Comme par exemple la notion d'argent, de vie et même les différents ennemis. Cependant pour des soucis de temps certaines choses ont dû être simplifiées comme la boutique ou le nombre de salles correspondant au monde du jeu.

2. Faits en désaccord avec le cahier des charges

Bien que nous fussions motivés à réaliser tout ce que nous avions prévu, il a été difficile pour nous de réaliser certaines choses initialement prévues. Néanmoins, celles-ci ne sont que mineures et ne nuisent pas au fonctionnement du jeu. Par exemple nous n'avons pas implémenté le fait qu'un ennemi puisse nous rendre de la vie une fois que le joueur l'a éliminé. Celui-ci nous donnera uniquement de l'argent qui pourra être utilisé dans une boutique. Aussi, nous n'avons pas implémenté de bouton rejouer à proprement parler. Mais il est possible de rejouer une partie en retournant au menu principal (grâce au bouton « continuer » en fin de partie) et en lançant une nouvelle partie.

XII. CONCEPTION

Dans cette partie nous justifierons les choix que nous avons dû réaliser tout au long du projet pour chaque fonctionnalité du jeu.

1. Sujet du projet

Les jeux vidéo étant pour nous l'un de nos loisirs, c'est tout naturellement que nous avons choisi de réaliser un jeu vidéo en 2D. L'utilisation au quotidien de jeux nous a facilité la réalisation de certaines fonctionnalités car nous savons ce qui est nécessaire dans une expérience vidéo ludique. De cette manière, certaines fonctionnalités étaient pour nous des évidences, comme un menu pause ou les contrôles.

2. Langage

Pour le langage aussi le choix fut évidant. Tout au long de notre cursus scolaire au sein de l'UPJV, il nous a été demandé de travailler en langage C. C'est pourquoi nous avons choisi ce langage même si après réflexions il aurait été plus simple de le faire dans un langage objet comme le Java ou le C++. Nous avons donc choisi notre langage pour l'affinité que nous avons avec celui-ci. Il en est de même pour les bibliothèques, ici la SDL2.

3. Direction artistique

Pour la direction artistique nous nous sommes mis d'accord sur un univers précis qu'est le rétro-futurisme. Deux couleurs se dégagent énormément dans cet univers ; le cyan et le violet. C'est de cette manière que nous avons trouvé notre charte graphique. Aussi quant à la direction artistique, nous avons décidé de réaliser des illustrations en *Pixel Art*. De cette manière il était beaucoup plus simple pour nous de dessiner des personnages ou décorations pour l'environnement du jeu.

4. Contrôles

Comme nous l'avons dit précédemment les contrôles du jeu étaient pour nous une évidence. C'est pourquoi nos choix se sont faits rapidement. Ainsi nous avons choisi les touches ZQSD ainsi que les touches directionnelles du clavier. Ces contrôles sont intuitifs et facilitent les déplacements. De même pour la souris nous avons choisi le clic gauche ou droit pour tirer. Pour les interactions nous avons choisi une touche régulièrement utilisée dans les jeux vidéo. Grâce à la touche E l'utilisateur pourra interagir avec la boutique. Il en est de même pour le menu pause, nous avons choisi la touche ECHAP. Nous avons aussi décidé d'ajouter un bouton, ici P, pour mettre en pause et reprendre la musique du jeu.

5. Environnement de travail

Dans le but de posséder un certain confort pour la réalisation de ce projet nous avons choisi un environnement familier pour le développement du projet. Ainsi pour le codage nous avons tous deux utiliser un IDE mais nous y reviendrons dans la partie propre au codage. Pour la réalisation des diverses illustrations nous avons utilisé la suite Adobe. Pour la musique nous avons découvert un site internet permettant la création de musique appelé BeepBox. Le site est très intuitif c'est pourquoi nous l'avons choisi.

XIII. CODAGE

Dans cette partie nous reviendrons sur le code que nous avons réalisé tout au long de la phase de conception. Nous parlerons d'abord de l'environnement que nous avons utilisé pour coder puis de la structuration du code.

1. Environnement

Comme nous l'avons dit précédemment nous avons tous deux utiliser un IDE pour mener à bien ce projet : l'un utilisant CodeBlocks l'autre Xcode. Ces IDE nous ont beaucoup aidé pour les fonctionnalités que chacun apporte. Par exemple les *include* se mettent automatiquement ou encore les détections d'erreurs ou de warnings sont beaucoup plus claires.

2. Structures utilisées

Pour réaliser un jeu en 2D en langage C il était indispensable d'utiliser une bibliothèque c'est pourquoi nous avons choisi la SDL2 pour sa simplicité. Cette bibliothèque nous permet de créer entièrement le jeu grâce à ses diverses fonctions et ce, de manière simple.

Pour structurer le code nous avons utilisé énormément de structure propre au langage C. Ces structures nous évitent de passer énormément de paramètres dans certaines fonctions et nous évitent la création de nombreuses variables.

En ce qui concerne la structure propre du code, elle s'articule de la façon suivante :

```
Créer FENETRE
Tant que FENETRE alors
    Affichage FENETRE
    Tant que JOUER alors
        Affichage JOUER
        Tant que PAUSE alors
            Affichage PAUSE
        Fin Tant que
    Fin Tant que
Fin Tant que
```

XIV. TEST

De manière à pouvoir tester notre jeu, nous établirons 3 scenarios. Ces 3 scenarios définiront la structure de cette partie portant sur le test du code.

1. Le joueur gagne la partie

Dans ce scenario, le joueur gagnera la partie. Il sera donc venu à bout de l'ennemi final. Dans sa quête de la victoire, il aura acheté deux améliorations, l'une lui permettant de tirer plus vite, l'autre de lui rajouter un peu de vie :

- Dans un premier temps le joueur arrive sur le menu principal :



- Ensuite il arrive dans la première salle :



- Il achète deux améliorations dans la boutique :



- Il combat le dernier ennemi :



- Et gagne la partie :



2. Le joueur perd la partie

Dans ce scenario, le joueur perdra la partie, il se sera fait éliminer par des ennemis lambda après avoir mis pause dans la troisième salle. Le joueur n'aura pas acheté d'améliorations et souhaitera rejouer au jeu.

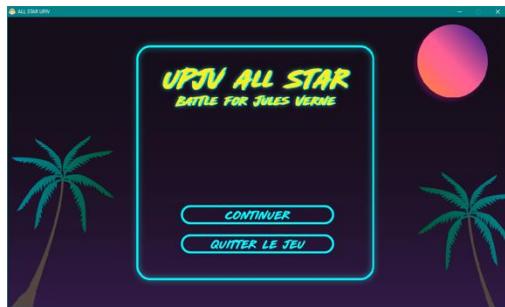
- Dans un premier temps le joueur arrive sur le menu principal :



- Ensuite il arrive dans la première salle :



- Il met en pause le jeu :



- Se fait tuer par les ennemis :



- Il perd la partie :



- Et souhaite rejouer :



3. Démonstration

Dans le but de montrer le fonctionnement du jeu nous avons réalisé une vidéo YouTube dans laquelle nous expliquons comment le jeu marche (cf. lien [Youtube](#)).

XV. BILAN

Après ces dix semaines d'analyse et conception nous pensons avoir réalisé un projet cohérent avec le cahier des charges initial. Nous avons implémenté toutes les fonctionnalités initialement prévues malgré les difficultés rencontrées.

1. Difficultés

Ce projet était une première pour nous, nous n'avions jamais réalisé de projet d'une telle ampleur. Nous avons donc fait face à de nombreuses difficultés que nous avons réussi à surmonté avec le temps.

La principale difficulté à laquelle nous avons dû faire face est la *deadline*. Il a été difficile pour nous de d'évaluer le temps que prendra chaque fonctionnalité et donc chaque tâche à réaliser. Nous avons donc dû négliger certaines fonctionnalités pour en privilégier d'autre.

Aussi, le travail en coopération a été un obstacle pour nous. Comme nous avons privilégié le langage C il était impossible pour nous de travailler chacun de n'autre coté sur des méthodes ou fonction propre à un même objet. Avec le recul il aurait été plus judicieux d'utiliser un langage objet comme le langage Java ou encore le langage C++.

Nous avons aussi rencontré des difficultés mineures propre au code que nous avons produit. En effet, il nous est arrivé de rester bloqué plusieurs heures ou jours sur certaines coquilles qui faisaient planter le jeu entier.

2. Améliorations

En prenant du recul sur le travail que nous avons fourni il est évident que nous aurions pu améliorer certaines choses. Comme nous l'avons dit précédemment, il aurait été possible pour nous de changer de changer le langage initialement prévu par un langage orienté objet, comme le Java ou le C++. De plus ces langages bénéficient de bibliothèques adéquates à la création de jeux vidéo.

3. Bénéfices

La réalisation de ce projet nous a apporté beaucoup de choses que nous n'avions pas. Il nous a permis de nous familiariser avec la création de projet à envergure professionnelle et de comprendre les enjeux et difficultés que cela pouvait engendrer. Même si cela fut fastidieux, nous avons énormément appris sur la rédaction d'un projet à l'écrit ; la création d'un cahier des charges, la rédaction d'un rapport. Aussi nous avons beaucoup appris dans le

langage de programmation que nous avons choisi. Nous sommes maintenant en mesure, grâce à ce projet, de réaliser dans son intégralité un projet quasi professionnel.

XVI. CONCLUSION

Après ces dix semaines des conceptions nous sommes en mesure de présenter notre projet de jeu vidéo en 2D. Le jeu est totalement fonctionnel et présente toutes les fonctionnalités initialement prévues dans le cahier des charges. Même si nous avons rencontré quelques difficultés durant la réalisation de celui-ci, nous sommes fier de ce que nous avons produit et fier de vous présenter notre jeu. Nous avons tiré énormément de positif de la réalisation de ce projet de par les connaissances qu'on en a tiré mais aussi l'expérience acquise.

XVII. BIBLIOGRAPHIE

Shoot em up[en ligne], Wikipedia, 2020,
https://fr.wikipedia.org/wiki/Shoot_%27em_up

Enter the Gungeon[en ligne], Wikipedia, 2019,
https://fr.wikipedia.org/wiki/Enter_the_Gungeon

Hotline Miami[en ligne], Wikipedia, 2020,
https://fr.wikipedia.org/wiki/Hotline_Miami

Gameplay [en ligne], Wikipedia, 2020,
<https://fr.wikipedia.org/wiki/Gameplay>

Pixel Art [en ligne], Wikipedia, 2020,
https://fr.wikipedia.org/wiki/Pixel_art

Roguelike [en ligne], Wikipedia, 2020,
<https://fr.wikipedia.org/wiki/Roguelike>

The Binding of Isaac [en ligne], Wikipedia, 2020,
https://fr.wikipedia.org/wiki/The_Binding_of_Isaac

NBA All Star 2014 [en ligne], Wikipedia, 2018,
https://fr.wikipedia.org/wiki/NBA_All-Star_Game_2014

Wave Attack Font [en ligne], Dafont, 2019,
<https://www.dafont.com/wave-attack.font>

VCR OSD [en ligne], Dafont, 2015,
<https://www.dafont.com/vcr-osd-mono.font>

BeepBox [en ligne]
<https://beepbox.co>