

Regnorm - Cahier des charges

Groupe PizzaBièreCafé:

CHEYNET Samuel
DUJARDIN Tristan
MICHALET Alix
PINGLOT Thomas

January 2021



Table des matières

1	Introduction	2
1.1	Origines du groupe	2
1.2	Présentation du Groupe	3
1.2.1	Samuel Cheynet	3
1.2.2	Tristan Dujardin	3
1.2.3	Thomas Pinglot	3
1.2.4	Alix Michalet	4
2	Origine et nature du projet	5
3	Objet de l'étude	6
3.1	Nos objectifs	6
3.2	Moyens Utilisés	6
4	Projet Technique	7
4.1	Gameplay	7
4.2	Terrains	7
4.3	Physique	7
4.4	Caméra	8
4.5	Menus	8
4.6	Réseau	8
4.7	Campagne solo	8
4.8	Collectibles	9
4.9	Ambiance Graphique et sonore	9
5	État de l'art	10
6	Découpage du projet	12
6.1	Répartition	12
6.2	Avancement au cours des soutenances	13
7	Conclusion	14

1 Introduction

1.1 Origines du groupe

Notre groupe ne s'est pas formé dès le début de l'année comme beaucoup d'autres groupes. Il n'y avait au début que deux personnes, Samuel et Tristan. À la suite d'une annonce sur le discord de la promotion, Alix est devenu un membre du groupe puis Thomas s'est présenté car il n'avait pas de groupe et était intéressé par le projet. Nous ne nous connaissons donc que depuis peu pour beaucoup, cependant nous pensons que cela peut être une force car le groupe met ainsi plus de pression sur ses membres qui ne veulent pas décevoir les autres et nous pouvons ainsi être bien plus productifs. Pour ce qui est du nom du groupe nous avons décidé de trouver un nom qui réunit nos valeurs communes, PBC (PizzaBièreCafé). Ce nom, nous l'espérons sera la définition même de nos soirées de travail sur le projet!¹

1. nous sommes très fières de notre nom

1.2 Présentation du Groupe

1.2.1 Samuel Cheynet

J'ai comme beaucoup de mes camarades toujours eu beaucoup de plaisir à découvrir de nouveaux jeux vidéo, leurs fonctionnalités, leurs nouveautés, et de manière générale les mondes qu'ils créent. Ce n'est qu'en entrant au collège que j'ai commencé à me renseigner sur les métiers derrière notamment en découvrant les commandes dans le jeu "Minecraft" qui permettent de créer ses propres règles. Mais aussi en découvrant les graphismes via le pixel art. C'est pourquoi je me suis tout naturellement tourné vers l'informatique et ce dès le lycée avec des sites tels que « open Classroom » en apprenant le Python puis le HTML et le CSS. C'est pourquoi, maintenant que l'occasion se présente de créer son propre jeu vidéo je n'ai pas hésité, que ce soit pour gérer la direction artistique et réutilisant et perfectionnant mes compétences de pixel art, mais aussi en apprenant à gérer les règles de physique dans le jeu en découvrant la partie cachée de l'iceberg que sont le code et toutes les choses qui nous paraissent anodines quand on joue à un jeu car évidentes pour notre cerveau (règles de physique dans un jeu par exemple).

1.2.2 Tristan Dujardin

Je suis depuis toujours un grand fan de jeux vidéos, en particulier ceux avec un éditeur de niveau, où j'ai pu avoir une première expérience de game design. J'étais donc très enthousiaste quand j'ai entendu parler du projet du semestre 2 où je compte à présent réaliser ce qui m'était impossible à faire dans ces éditeurs : créer quelque chose en partant de rien. J'ai commencé à programmer il y a plusieurs années, en C++, mais j'ai très rapidement perdu ma motivation à cause de la complexité du langage. Mais grâce aux cours d'ISN en terminale et au langage Python, j'ai redécouvert la programmation et j'en suis resté passionné depuis. J'ai déjà essayé de coder un jeu de combat en 2d pour le projet d'ISN, et même si ce projet a été abandonné, j'ai beaucoup appris et je suis fier de ce que j'y ai réalisé. Maintenant, je compte bien amener notre projet actuel à aboutissement. Les parties qui m'intéressent le plus dans ce projet sont l'IA et le Réseau, dont je serai responsable lors de ce projet.

1.2.3 Thomas Pinglot

Depuis tout petit j'ai très vite aimé les jeux vidéos et en grandissant j'ai toujours voulu un jour créer un jeu. J'ai commencé la programmation en terminal avec

la spé ISN où on y faisait du python très basique. J'ai donc vraiment commencé à programmer au début d'EPITA, et j'ai fait un jeu l'année dernière lors de mon projet S2. Je compte bien mettre à profit mon expérience du projet de l'année dernière afin que celui-ci soit encore meilleur. Ayant déjà fait un site web je m'occuperai de faire celui de notre projet, je m'occuperai également des menus de notre jeu et j'aiderai à coder les objets de déplacement mais aussi les armes de notre jeu.

1.2.4 Alix Michalet

J'ai toujours été intéressé par l'informatique depuis tout petit et plus particulièrement par les jeux vidéos, je suis ce qu'on peut appeler un "geek". J'ai passé énormément de temps à jouer à tous types de jeux vidéos et en créer un a toujours fait partie de mes fantasmes. Petit je passais mon temps à jouer sur diverses plateformes mais plus particulièrement téléphone, tablette, etc. avant de commencer à jouer sur PC. Grâce à tout ce temps passé à jouer et grâce aux centaines de jeux que j'ai pu tester j'espère pouvoir reprendre les meilleurs aspects de chacun et les implémenter (tout en les améliorant) dans notre jeu. En intégrant l'EPITA je m'étais donné pour objectif de recréer le jeu « blastron » et c'est donc pour cela que l'on a décidé de faire un jeu « worms like » : un jeu d'artillerie. J'ai toujours eu une imagination débordante et une envie de modéliser le jeu et plus particulièrement le lore afin de lui donner une image qui me serait propre c'est pour ça que je suis responsable du lore, de la campagne et suppléant pour les graphiques. Dans ce projet je m'occupe également en tant que suppléant de l'IA (intelligence artificielle), de l'OST/BO (Original Soundtrack/Bande Originale) et des effets visuels de destruction.

2 Origine et nature du projet

A l'origine, le groupe qui n'était composé que de Tristan et Samuel n'avait pas d'idée précise pour le projet si ce n'est d'utiliser Unity car ils en avaient entendu beaucoup de bien et ne voulaient absolument pas avoir affaire au Caml (important traumatisme). Ce n'est qu'après l'arrivée d'Alix dans le groupe que nous nous sommes orientés vers un jeu de type Artillerie (worms-like : inspirés du jeu Worms, Pvp ou PvE au tour par tour dans une arène destructible). Étant donné que ce type de jeu nous permet d'explorer beaucoup de possibilités allant de la physique de destruction jusqu'au multijoueur le projet sera très enrichissant pour le groupe. Compte tenu du fait que nous soyons tous des « gamers »/ « geeks » nous nous sommes très rapidement mis d'accord sur le fait de créer un jeu vidéo. Ce projet est donc l'occasion parfaite pour nous de faire un jeu dans le cadre d'un « projet professionnel ». Nous sommes tous très enjoués à l'idée de développer quelque chose qui a pour seul but de divertir.

3 Objet de l'étude

3.1 Nos objectifs

Pour ce projet, nous comptons avant tout nous améliorer en programmation mais également réaliser un fantasme commun à tous les membres du groupe en développant un jeu vidéo dans son entièreté.

Au travers de ce projet, nous allons devoir apprendre beaucoup de choses, que ce soit travailler en équipe de façon organisée ou bien résoudre par nous-mêmes tous les problèmes auxquels nous serons confrontés. Pour cela, nous allons devoir apprendre à nous entraider et à travailler selon les préférences, les forces et les faiblesses de chacun. Finalement, nous serons également forcés de sacrifier encore plus de temps libre (et de sommeil !) pour ce projet.

Malgré tout cela, nous comptons bien créer quelque chose dont on sera fiers, quelque chose qui aura valu tous ces sacrifices, et un jeu sur lequel nous pourrions prendre du plaisir en y jouant de temps à autres.

3.2 Moyens Utilisés

Nous comptons également apprendre à utiliser plusieurs nouveaux outils qui nous serviront sans doute beaucoup plus tard :

Pour gérer l'avancement du projet, s'organiser et rassembler nos travaux nous utiliserons GitHub.

Pour les graphismes, nous créerons les textures, les sprites et les animations en pixelart à l'aide du logiciel Asperite.

Pour rédiger ce cahier des charges et tous les rapports de soutenances, nous utiliserons le site Overleaf, qui va nous permettre à tous de modifier le même fichier en \LaTeX . Enfin, pour le moteur du jeu, nous utiliserons Unity, que nous avons préféré à Godot pour plusieurs raisons :

- L'implémentation du C# qui y est bien plus ancienne que sur Godot, donc bien plus documentée.
- Le High Level API (ou HLAPI) qui est un très bon outil pour gérer toute la partie réseau.
- Samuel et Tristan voulaient utiliser Unity, et Thomas qui a déjà de l'expérience nous a également conseillé ce moteur.
- Unity étant un moteur avancé, il nous permettra de simplifier la création des bases du jeu et de nous concentrer sur les parties qui nous intéressent, comme créer des collectibles, un système de multijoueur, des IA et une campagne solo.

4 Projet Technique

4.1 Gameplay

La gameplay de Ragenorm sera basé sur les combats solo et multijoueurs (entre 2 et 4 joueurs). Dans un combat, les joueurs pourront effectuer des actions dans un système de tour par tour. Ces actions seront l'attaque, le déplacement, la récupération d'armes dans des caisses sur le terrain et l'utilisation des capacités. (le tour d'un joueur se termine à la fin du temps imparti ou bien quelques secondes après qu'il ai lancé son attaque afin qu'il puisse se repositionner, ainsi, déplacements et utilisation des capacités n'impacteront pas le temps du tour) Lors d'une attaque, les joueurs infligent des dégâts aux autres selon l'arme utilisée, et gagnent un nombre de points correspondants aux dégâts infligés (un nombre de points bonus sera attribué si la cible est tuée lors de l'attaque ou si elle est tuée par des dégâts continus). Une partie se terminera quand le timer, réglable à la configuration de la partie, est écoulé, et le joueur ayant le plus de points à ce moment là remporte la partie.

4.2 Terrains

Les terrains sont l'endroit où évolueront les joueurs lors d'une partie, ils seront composés de quatre éléments :

- Le background : une image fixe qui sera placée derrière tous les autres éléments.
- Le terrain indestructible : une partie du terrain sur laquelle les joueurs pourront marcher même après la destruction partielle ou totale du terrain destructible.
- Le terrain destructible : une partie du terrain sur laquelle les joueurs pourront marcher tant qu'elle est intacte, mais elle peut être détruite par certaines armes et compétences.
- Le fond de terrain : il a la même forme et place que le terrain destructible, mais il est affiché derrière lui, ce qui fait qu'il ne sera montré qu'aux endroits détruits du terrain destructible indiquant que les joueurs peuvent le traverser.

4.3 Physique

La physique du jeu pourra se diviser en 2 parties, la physique des joueurs et la physique du terrain. Lors de leur tour de jeu, les joueurs pourront se déplacer sur les terrains intacts en marchant et en sautant mais ils ne pourront pas traverser les hitbox des autres joueurs et des caisses d'armes (ils pourront passer par dessus en

sautant et/ou les pousser). Lors des attaques d'un joueur avec certaines armes, le terrain destructible sera détruit dans la zone d'impact de l'attaque.

4.4 Caméra

Il y a La caméra sera pour ce projet une simple caméra en 2D qui suivra les déplacements horizontaux des joueurs sur le terrain. Elle fera également bouger le background plus lentement que les joueurs et les parties du terrain au premier plan pour donner un effet simple de perspective.

4.5 Menus

Avant tout, les menus devront être intuitifs et faciles d'utilisation. Le joueur pourra facilement s'en servir à l'aide d'une manette ou d'un clavier et d'une souris. Le menu principal devra donner accès aux modes de jeu solo et multijoueur, l'accès à l'inventaire du compte et aux paramètres du jeu.

Le menu multijoueur devra donner la possibilité d'héberger une partie en local ou en réseau et d'en rejoindre une via une IP.

Lors d'une partie, le menu sera accessible par la touche ESC du clavier ou START d'une manette, il donnera accès aux paramètres ainsi qu'à un bouton pour quitter le menu de pause et un autre pour revenir au menu principal.

4.6 Réseau

Pour le réseau, le système de multijoueur fonctionnera sur un système de Pair-2-Pair (P2P), avec un joueur "hôte" qui hébergera une instance de serveur en plus du client, et entre un et trois autres joueurs qui pourront le rejoindre à l'aide de leurs adresses IP. L'hôte se chargera de configurer la partie et de la démarrer une fois que les autres joueurs l'auront rejoint.

4.7 Campagne solo

La campagne solo sera divisée en 9 Chapitres, chacun se jouant avec un personnage spécifique aux compétences diverses qui seront ensuite jouables en partie multijoueur. Chaque chapitre se composera d'un tutoriel apprenant au joueur les bases du personnage et de ses compétences spécifiques, d'un niveau intermédiaire et d'un niveau final avec un boss de fin de Chapitre.

Dans les niveaux intermédiaires les joueurs seront confrontés à plusieurs IA avec

une difficulté croissante selon les Chapitres. Dans les niveaux finaux, le joueur devra vaincre un IA de "boss" qui aura une barre de vie affichée. Certains niveaux finaux auront des spécificités de gameplay et ne se composeront pas exclusivement d'un boss.

4.8 Collectibles

Les collectibles et l'inventaire des joueurs seront stockés dans les fichiers du client. Leur fonctionnalité sera de témoigner de l'avancement d'un joueur dans la campagne solo et d'être utilisés lors de combats en multijoueur. Lors d'une partie en réseau, chaque joueur aura accès à son propre arsenal d'inventaire, alors que dans une partie locale, tous les joueurs joueront sur le même inventaire du client local prédéfini dans les paramètres de la partie.

Ces collectibles seront composés de personnages, de récompenses et d'armes. Ces dernières se gagneront selon une probabilité de drop à la fin de chaque niveau lors de la campagne solo. Les personnages et récompenses seront débloqués après chaque chapitre (chaque chapitre permettra de débloquer un personnage et certains types d'armes (d'une puissance croissante selon le niveau des chapitres) que vous pourrez de nouveau obtenir sous forme de récompenses à la fin de vos parties multijoueurs en fonction de votre progression). Et oui, afin d'avoir un arsenal suffisant pour effrayer vos adversaires il vous faudra jouer en campagne ! Mais pour avancer dans la campagne il vous faudra également jouer en multijoueur afin d'avoir plus d'armes.

4.9 Ambiance Graphique et sonore

Pour les graphismes du jeu, nous tenons à les faire nous-mêmes, c'est pourquoi l'entièreté du jeu sera faite selon un style pixelart en 2D. Les terrains seront fait avec des tiles, tandis que les personnages et armes auront chacun un sprite. Des effets sonores seront également incorporés lors de certaines actions, comme les sauts et les coups infligés aux adversaires.

5 État de l'art

Notre jeu Regnorm est un jeu de type artillerie également appelé « worms-like » à cause du jeu le plus connu dans cette catégorie : Worms. Cependant, le premier jeu de ce type n'est pas Worms mais Artillery Duel dans les années 1980. Les jeux ayant fait le succès de cette catégorie sont : Worms, Gunbound, Hedgewars, Hogs of War et finalement Redcon. Ces jeux ont pour la plupart les mêmes qualités qui sont celles qui correspondent à la catégorie : ils sont simple à prendre en main car il n'y a pas énormément de commandes différentes (déplacement, tir et changement d'arme/compétences), ils sont très ludique sans pour autant être d'une violence inouïe et sont tous en multijoueur.

Le jeu Hogs of War (2000) se démarque principalement des autres par sa caméra (vue à la troisième personne alors qu'en temps normal, les jeux d'artillerie sont en 2D avec une vue de profil d'ensemble) et donc en conséquence par son gameplay qui ne sera pas « à plat ». Ce changement est tout autant positif que négatif : il apporte une nouvelle façon de jouer à un jeu d'artillerie mais en même temps, cela dénature cet aspect 2D qui est la marque de fabrication des jeux d'artillerie type.

Le jeu Hedgewars (2006) quant à lui apporte un aspect bien plus enfantin à la catégorie de jeu en développant des personnages et un univers quasiment féérique. (concept très inspiré de Worms mais bien plus poussé avec des personnages semblables à Kirby et des environnements très fantaisistes avec des taupes, des aurores boréales et des fleurs) Pour sa part, le jeu Gunbound (2003) est le premier à insérer des « effets de météo » avec un vent pouvant tourner et déviant les projectiles, un grand soleil n'ayant pas d'impact sur ces derniers ou bien encore l'orage, la tempête etc.. Ces ajouts sont très intéressants et rendent le jeu un peu plus réaliste par rapport à l'environnement tout en compliquant et en rajoutant du gameplay.

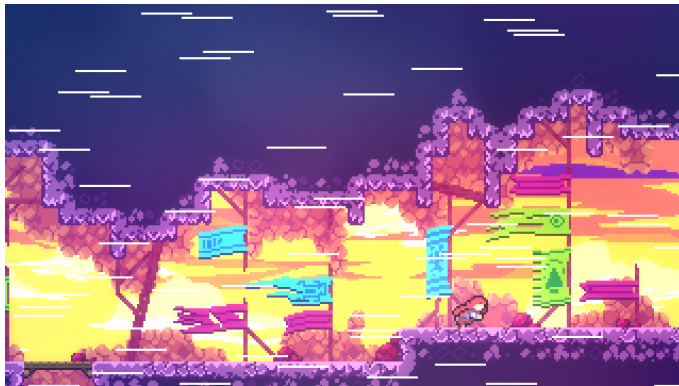
Redcon (2016) est, lui, bien plus récent et a apporté un aspect très stratégique aux jeux d'artillerie. Celui-ci ne se base pas sur les mouvements d'un joueur et la possibilité de tuer ses adversaire en leur tirant simplement dessus mais sur la construction d'une forteresse armée afin de détruire celle de l'adversaire.

Notre jeu à nous sera bien plus inspiré par les jeux Blastron et Worms tant dans le gameplay que dans le Lore et les mécaniques/physique.



FIGURE 1 – Gameplay du jeu "Blastron"

Pour notre jeu nous avons choisis d'utiliser le pixelart pour un certains aspect rétro dans le jeu mais également afin de pouvoir créer notre propre univers avec notre vision sans pour autant avoir le moindre talent en illustration ou en dessin. Nos inspirations majeures pour les graphismes sont les jeux qui nous ont éduqué à l'art du gaming depuis notre tendre enfance comme Terraria, Celeste ou encore Stardew Valley.

Graphiques des jeux "Celeste" et
"Terraria"

6 Découpage du projet

6.1 Répartition

Répartition	Samuel	Thomas	Tristan	Alix
Game design	Suppléant			×
Graphique	×			Suppléant
Menus		×	Suppléant	
Site Internet	Suppléant	×		
Réseau			Suppléant	×
IA			×	Suppléant
Physique/mouvement	×		Suppléant	
Item déplacement	×	Suppléant		
Campagne	Suppléant			×
Arme			×	Suppléant

6.2 Avancement au cours des soutenances

Répartition	1ère soutenance	2ème soutenance	3ème soutenance
Game design	25%	50%	100%
Graphique	25%	50%	100%
Menus	50%	100%	×
Site Internet	50%	80%	100%
Réseau	25%	60%	100%
IA	×	50%	100%
Physique/mouvement	50%	100%	×
Item déplacement	20%	60%	100%
Campagne	×	50%	100%
Arme	20%	60%	100%

7 Conclusion

Nous avons décidé de réaliser un jeu vidéo 2D en nous inspirant d'un jeu existant déjà, "Worms", car cela nous semble intéressant de travailler sur un jeu que l'on connaît et que l'on trouve amusant, mais aussi car il est faisable avec nos capacités actuelle et celles que nous développerons dans l'année. Nous pensons que ce projet peut nous apporter beaucoup, que ce soit pour nous apprendre à travailler en groupe sur un projet informatique, mais aussi sur comment développer nos idées pour aboutir à un projet. Nous allons utiliser des ressources adaptées à la création d'un jeu en C#, tel que "Unity". Nous ferons tout notre possible pour respecter les deadline en essayant potentiellement de prendre de l'avance sur nos prévisions.