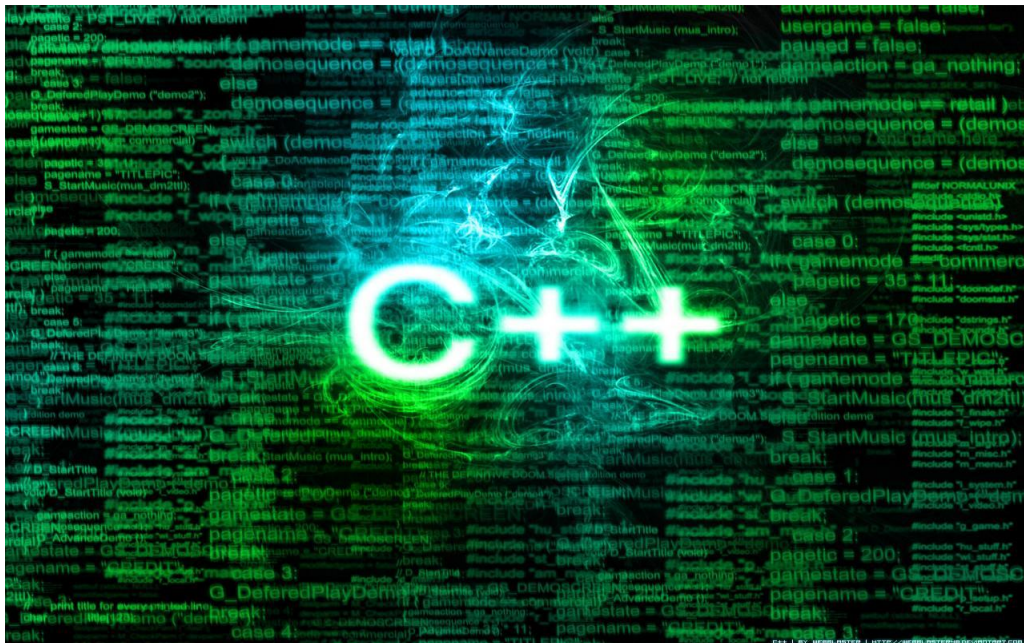

Rapport de Projet : L'Apocalypse



HÉLOÏSE VIOSSAT

MAXIME LAFONT-TREVISAN

2020-2021

Table des matières

1	Introduction	2
2	Règles du jeu	2
2.1	Choix du mode	2
2.2	Les météorites	3
2.3	Les Supers personnages	3
3	Nos classes et fonctionnalités	3
3.1	Diagramme UML	3
3.2	La classe météorite	4
3.3	La classe personnage	5
3.3.1	Les humains	5
3.3.2	Les extra-terrestres	6
4	L'exécution	6
5	Conclusion	6

1 Introduction

Dans le cadre de ce projet de C++ nous avons dû réfléchir à un sujet dans le thème du **Monde d'après**. Nous avons choisi de réaliser un jeu dans lequel le joueur incarne un personnage et doit éviter des météorites.

Nous avons décidé de faire un projet utilisant principalement la bibliothèque SDL2 pour l'affichage graphique de notre jeu. Nous avons ainsi pu nous familiariser avec cette bibliothèque et appris l'affichage graphique et la gestion d'événements.

Ce projet a également été l'occasion de mettre en pratique les notions de programmation objet et de hiérarchie de classes.

2 Règles du jeu

Le principe du jeu est assez simple et intuitif. Il s'agit de déplacer un personnage, représenté par un carré de couleur, afin d'éviter ou d'attraper des météorites qui tombent du ciel, c'est à dire du haut de la fenêtre.

2.1 Choix du mode

Avant de débiter le jeu, le joueur doit choisir entre deux modes possibles :

- Le mode Humain dans lequel le joueur incarne un humain qui se déplace latéralement sur le bas de la fenêtre, représentant le sol, et peut évoluer en SuperHumain ;
- Le mode Extra-Terrestre dans lequel le joueur incarne un extra-terrestre qui a les mêmes capacités que l'Humain et peut évoluer en SuperExtraterrestre.

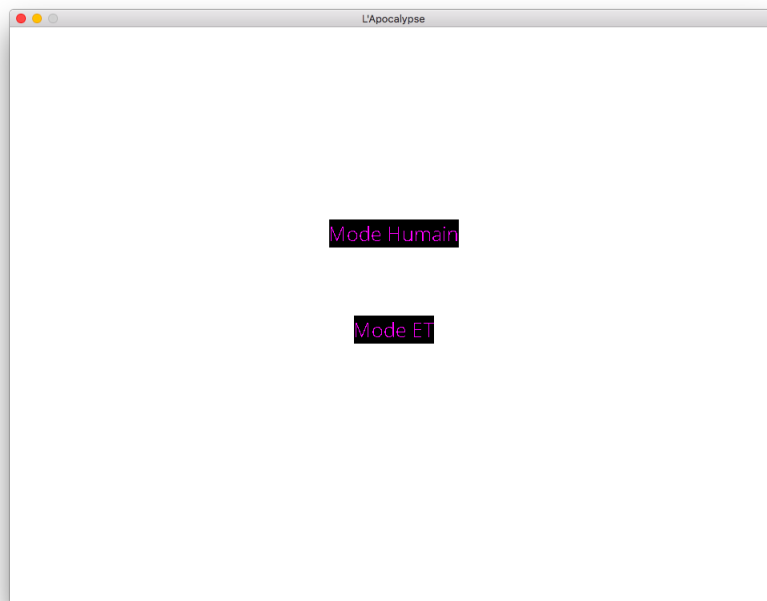


FIGURE 1 – Menu mode

2.2 Les météorites

Une fois ce choix fait, le jeu débute. Le joueur possède à l'origine trois vie et joue avec le personnage de base du mode sélectionné. Il déplace le personnage à l'aide des flèches du clavier. Lorsqu'il entre en contact avec une météorite plusieurs conséquences sont possibles selon la couleur de la météorite. Les météorites rouges sont à éviter car elle font perdre une vie. Les météorites bleues et vertes, beaucoup plus rares, sont à attraper. En effet, les premières permettent au personnage d'évoluer et ainsi d'acquérir un pouvoir. Les secondes ajoutent une vie au personnage.

Le jeu se termine lorsque le personnage n'a plus de vie.

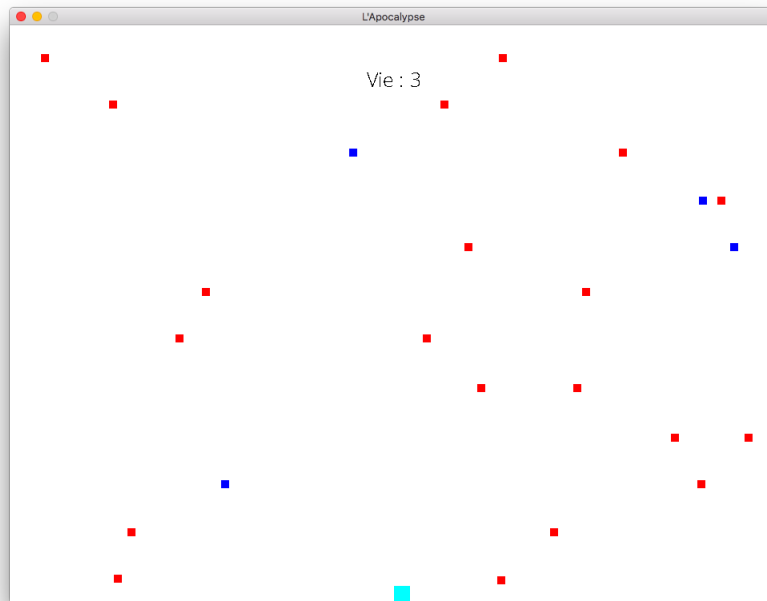


FIGURE 2 – Le jeu en action

2.3 Les Supers personnages

Lorsque le joueur attrape une première météorite bleue, son personnage évolue et devient un SuperHumain ou SuperET selon le mode choisi. Il peut alors utiliser son pouvoir en appuyant sur la touche "A". Le pouvoir est alors actif pendant 15 secondes et après sa désactivation il devient réactivable au bout de 20 secondes (évidemment ces valeurs peuvent être changées).

Chaque Super Personnage à un pouvoir différent. Le SuperHumain possède un bouclier qui le rend invincible et le SuperET a le pouvoir de ralentir les météorites et voler (c'est à dire avoir la capacité de se déplacer en haut et en bas).

3 Nos classes et fonctionnalités

3.1 Diagramme UML

Les deux catégories d'objets dans notre jeu sont les météorites et les personnages, ce qui nous a conduit à implémenter deux classes : **meteorite** et **personnage**. La première représente les météorites qui tombent sur le

personnage, et la seconde, le personnage, contrôlé par le joueur, et qui doit éviter les météorites. Le personnage peut être de différents types ce qui se manifeste par différentes classes héritant de **personnage** que nous détaillerons par la suite.

Le diagramme de classes suivant représente les différentes classes de notre jeu.

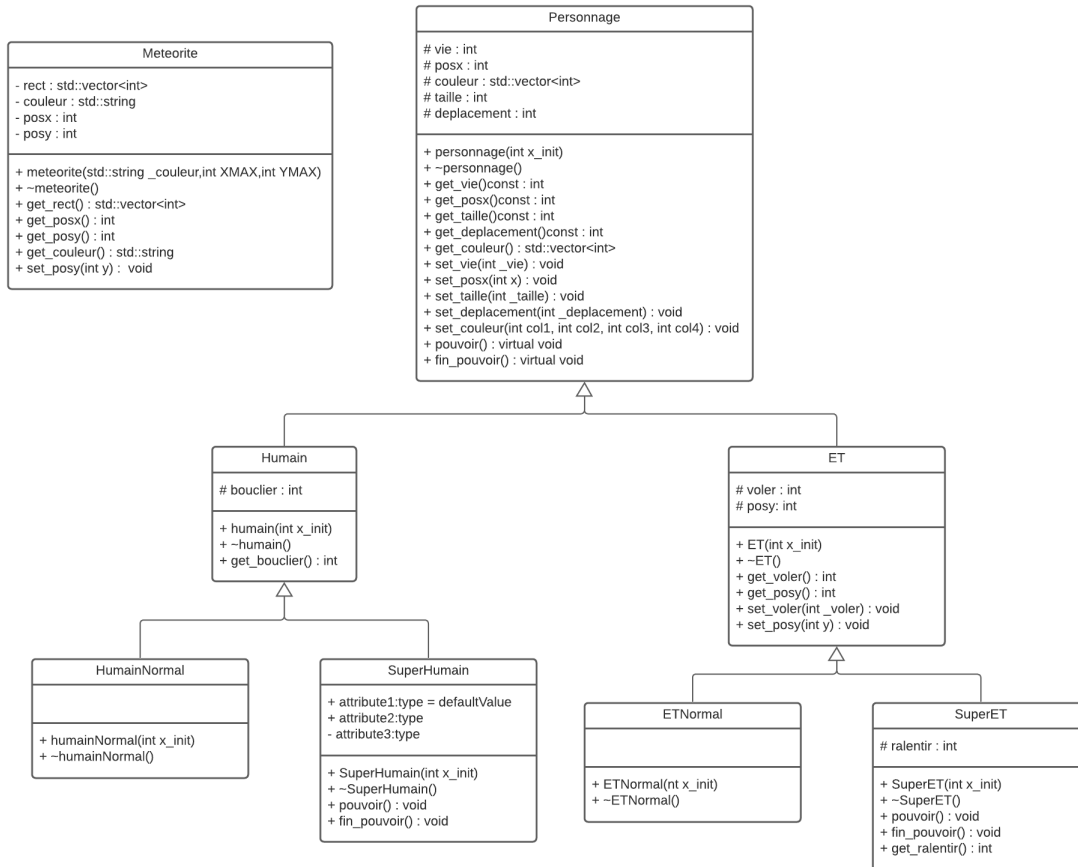


FIGURE 3 – Diagramme de Classes

3.2 La classe météorite

La classe *meteorite* représente les météorites du jeu. Leur rôle est de tomber verticalement du haut de la fenêtre qui représente le ciel, vers le bas qui matérialise le sol. Lorsqu'elles entrent en contact avec le personnage, plusieurs conséquences sont possibles selon la météorite concernée.

Les attributs de la classe **meteorite** sont :

- *std::vector<int> rect*, un vecteur d'entiers nommé *rect*. Ce vecteur correspond au code d'une couleur compris par la bibliothèque *sdl2*. Le premier nombre correspond au rouge, le second au vert, le troisième au bleu et le quatrième à l'opacité. Ce code sera utilisé pour définir la couleur du rectangle représentant la météorite lors de l'affichage graphique.
- *std::string couleur*, une chaîne de caractère correspondant à la couleur de la météorite qui détermine son type.

- *int posx*, un entier correspondant à la position sur l'axe des abscisses de la météorite. L'extrémité gauche de la fenêtre correspond au 0, puis en se déplaçant vers la droite on se déplace vers les x positifs.
- *int posy*, un entier correspondant à la position sur l'axe des ordonnées de la météorite. L'extrémité supérieure de la fenêtre correspond au 0, puis en se déplaçant vers le bas on se déplace vers les y positifs.

Le constructeur de la classe **meteorite** prend comme arguments une chaîne de caractère correspondant à la couleur de la météorite et les dimensions de la zone d'apparition des météorites. Trois couleurs sont possibles :

- "rouge" pour les météorites normales. Celles-ci ont pour effet de faire perdre une vie au personnage lorsque que ce dernier en touche une.
- "vert" pour les météorites qui donne une vie au personnage.
- "bleu" pour les météorites qui font évoluer le personnage et lui permettent ainsi d'avoir un pouvoir.

La position initiale sur l'axe x de la météorites est définie aléatoirement. Sa position en x peut aller de 0 à XMAX, XMAX étant la largeur totale de la fenêtre. Sa position en y est initialisée à 0, c'est à dire en haut de la fenêtre.

Cette classe possède des getters pour chacun des quatre attribut et un setter pour la position en y, l'attribut *posy*.

3.3 La classe personnage

La classe **personnage** correspond au personnage du jeu, matérialisé par un caré de couleur. Le joueur peut déplacer son personnage à l'aide des flèches afin d'éviter ou d'attraper les météorites. Il existe deux types de personnages, les humains et les extra-terrestres, eux mêmes distinguables en deux sous catégories, normaux ou supers.

La classe **personnage** regroupe les attributs communs à tous les personnages : le nombre de vies (*vie*), la position horizontale du personnage (*posx*), la couleur contenue dans un vecteur d'entiers (*couleur*) et sa vitesse de déplacement sous forme d'un entier (*deplacement*).

Cette classe possède de plus les fonctions *get* et *set* pour chacun des attributs et deux fonctions virtuelles : *virtual void pouvoir()* et *virtual void fin_pouvoir()*. ces deux fonctions correspondent aux différents pouvoirs des personnages selon leur types.

3.3.1 Les humains

La classe **humains** hérite de la classe **personnage** car il s'agit d'un type de personnages. Les humains possèdent, en plus des attributs de la classe humain, un bouclier. Il s'agit d'un entier qui prend la valeur 1 s'il est activé, 0 si non. Il est inactivé à l'initialisation du jeu et peut être activé par la fonction *pouvoir* de la classe **SuperHumain**.

Deux classes héritent de celle-ci : les classes **HumainNormal** et **SuperHumain** qui correspondent aux types HN et SH.

La classe **HumainNormal** ne possède pas d'attribut en plus mais à une couleur correspondant au turquoise et une taille de 20. Il s'agit de la forme de base de l'humain.

La classe **SuperHumain** comporte les fonctions *pouvoir* et *fin_pouvoir*. La première permet de donner la valeur 1 au bouclier afin de l'activer, la seconde l'inactive en lui donnant la valeur 0. Le bouclier, une fois activé, rend invincible le personnage. Seul les SuperHumains possèdent un bouclier.

Cette classe correspond à la version évoluée de l'humain. Celui-ci possède désormais un pouvoir activable par le joueur à l'aide de la touche "A", qui dure 15 secondes et que le joueur pourra réactiver au bout de 20 secondes.

Le SuperHumain est plus petit que l'humain (taille = 10) ce qui lui permet d'éviter plus facilement les météorites, et a une couleur orange.

3.3.2 Les extra-terrestres

La classe **ET** hérite elle aussi de la classe **personnage** et représente les extra-terrestres. Les extra-terrestres se distinguent des humains par leur pouvoir et leur capacité à voler, c'est à dire à se déplacer vers le haut et le bas. La classe **ET** possède donc deux attributs en plus : *voler*, un entier correspondant au déplacement vertical de l'ET, et un entier *posy* qui prend la valeur de la position vertical de l'ET.

A l'instar de la classe **humain**, deux classes filles héritent de la classe **ET**, les classes **ETNormal** et **SuperET** correspondant aux types EN et SE.

La classe **ETNormal** est caractérisée uniquement par sa couleur qui correspond à du vert.

La classe **SuperET** possède deux méthodes pour activer et désactiver son pouvoir, *int pouvoir(int vitesse)* et *int fin_pouvoir(int vitesse)*, qui modifie la vitesse donnée en argument. En effet, ils ont le pouvoir de ralentir les météorites. Pendant le temps du pouvoir, les extra-terrestres peuvent également se déplacer vers le haut et le bas, ce qui correspond à voler. Ils peuvent ainsi attraper plus facilement les météorites vertes et obtenir de nouvelles vies. A la fin du pouvoir, l'extra-terrestre se retrouve à nouveau sur le sol et doit attendre de redevenir un Super Extra-terrestre pour pouvoir réactiver son pouvoir.

4 L'exécution

Afin de pouvoir exécuter notre jeu il faut installer la bibliothèque SDL2 qui permet l'affichage graphique du jeu.

Notre projet compile grâce à notre Makefile. Il y a une règle all et une règle clean. Tous les fichiers de classe sont des headers.

5 Conclusion

Ce projet a été l'occasion d'appliquer les notions étudiées ce semestre et d'appliquer le concept de hiérarchie de classes et programmation objet.

Il nous a de plus permis d'apprendre à utiliser la bibliothèque graphique SDL2.

Nous espérons que ce jeu vous amusera autant que le faire nous a amusé.