

UNIVERSITÉ POLYTECHNIQUE HAUTS-DE-FRANCE  
INSTITUT NATIONAL DES SCIENCES APPLIQUÉES

RAPPORT DE PROJET

# Moteur de jeu pour un jeu de plateforme 2D



Ethan MARLOT - 8 Avril 2022

# Table des matières

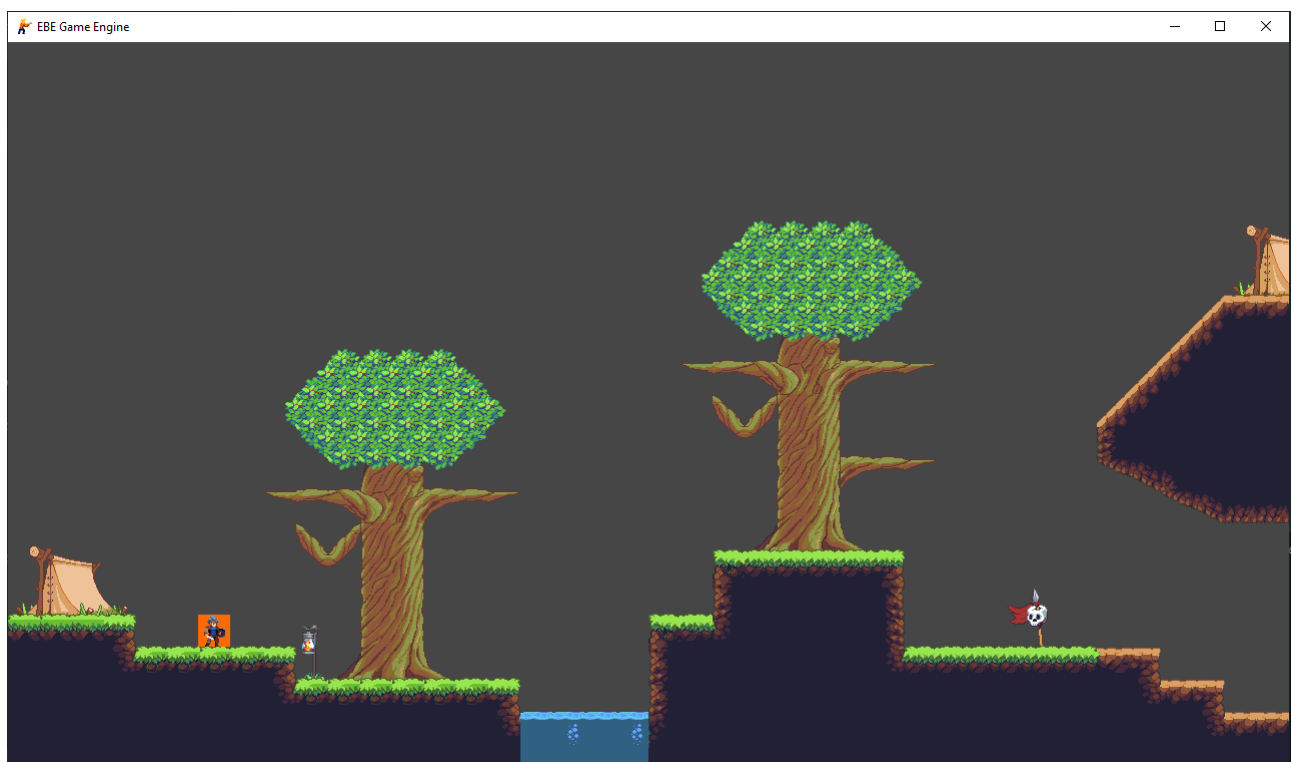
<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Analyse des besoins du projet</b>	<b>3</b>

# Chapitre 1

## Introduction

Dans le cadre de ce projet tuteuré de cette année 2022, j'ai choisi de développer un jeu de type "Plateformer-Shooter", similaire aux jeux d'arcade *Metal Slug*. Afin de réaliser ce projet, j'ai décidé de ne pas utiliser un moteur de jeu existant mais d'en créer une implémentation simple.

En l'état, voici l'apparence du jeu :



Nous avons une carte, ainsi qu'un personnage pouvant se déplacer librement sur cette dernière.

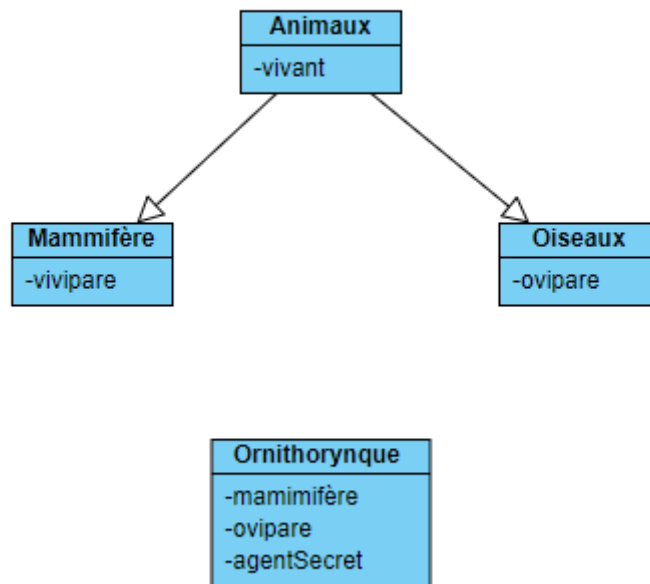
# Chapitre 2

## Analyse des besoins du projet

Tout d'abord, j'ai entrepris des recherches sur les moteurs de jeu. Lors de ces dernières, j'ai trouvé deux principaux modèles de moteur de jeu :

- Le modèle basé sur le paradigme de l'orienté objet ;
- Le modèle basé sur le paradigme de l'orienté donnée, ;

J'ai, dans un premier temps, pensé à choisir le paradigme orienté objet, étant donné que nous l'avons beaucoup travaillé lors de ces derniers semestres. Mais après réflexions, pour un moteur de jeu, l'orienté objet atteint ses limites. Prenons l'exemple simple suivant :



Dans cet exemple, nous avons une classe abstraite Animaux, dont héritent deux classes, Mammifères et Oiseaux. Les Mammifères sont vivipares tandis que les Oiseaux sont ovipares. Dans ce cas, quid de l'Ornithorynque, ce mammifère ovipare ? Il est vrai qu'en C++, il y a la possibilité d'héritages multiples, mais cela peut poser d'autres problèmes, comme dans l'exemple au dessus, si la classe Oiseaux a un attribut ailes, alors Ornithorynque ne devrait pas hériter d'Oiseaux.

C'est là où le paradigme de l'orienté donnée entre en jeu :

