



UFR SCIENCES ET TECHNIQUE  
UNIVERSITÉ MARIE ET LOUIS PASTEUR  
**PROJET**  
L3 INFORMATIQUE

---

# Jeu de plate-formes avec génération procédurale

---

RAPPORT DE PROJET

Kilian JELIC  
Laura JACQUESON  
Théo PARINEY

avril 2025

# Table des matières

1	Introduction . . . . .	3
2	Présentation du jeu . . . . .	4
2.1	Présentation générale . . . . .	4
2.2	Les objectifs du joueur . . . . .	4
2.2.1	Les écrous . . . . .	4
2.2.2	La sortie . . . . .	4
2.3	Les scènes . . . . .	4
3	Les blocks et entités du monde . . . . .	4
3.1	La définition dynamique des blocks . . . . .	4
3.2	Le stockage des blocks . . . . .	4
4	La génération procédurale . . . . .	4
5	Le personnage . . . . .	4
5.1	Les contrôles de base . . . . .	4
5.2	Le saut . . . . .	4
5.2.1	Le saut modulable . . . . .	4
5.2.2	Le double saut . . . . .	4
5.3	Le dash . . . . .	4
5.4	Le score et les vies . . . . .	4
6	La physique . . . . .	4
6.1	La gestion des collisions . . . . .	4
6.1.1	Les propriétés des blocks . . . . .	4
6.2	Calcul de la résistance de l'air . . . . .	4
7	Notre organisation . . . . .	4
7.1	La définition des tâches . . . . .	4
7.1.1	La répartition du travail . . . . .	4
7.2	Les outils utilisés . . . . .	4
7.2.1	Discord . . . . .	4
7.2.2	Google Docs . . . . .	4
7.2.3	Trello . . . . .	4
7.2.4	Git et Github . . . . .	4
7.2.5	IDE . . . . .	4
8	Conclusion . . . . .	5
8.1	Points d'amélioration . . . . .	5
8.1.1	Ce qu'on aurait voulu ajouter . . . . .	5
8.1.2	Wave Function Collapse . . . . .	5

8.1.3	Pathfinding amélioré . . . . .	5
8.1.4	La physique . . . . .	5

# Table des figures

# 1 Introduction

Dans le cadre de notre projet semestriel en L3 Informatique à l'Université Marie et Louis Pasteur, nous avons travaillé sur le développement d'un jeu de plate-formes avec génération procédurale en C++.

Les jeux de plate-formes sont un genre qui repose sur le contrôle d'un personnage avec des mécaniques comme les sauts et les obstacles, l'objectif étant généralement de rejoindre une sortie pour terminer le niveau. L'ajout de la génération procédurale introduit une caractéristique supplémentaire : plutôt que de concevoir chaque niveau à la main, un algorithme est chargé de créer les niveaux du jeu, permettant ainsi une expérience unique à chaque partie.

L'objectif principal de ce projet était de concevoir un jeu de plate-formes dont les niveaux seraient générés de manière procédurale, offrant ainsi une rejouabilité infinie. En parallèle, nous avons dû concevoir un jeu intégrant plusieurs mécaniques de jeu, incluant des déplacements, des sauts, un dash, ainsi qu'un objectif centré sur la récolte d'objets, tout en assurant la gestion des collisions et la physique du jeu.

Dans ce rapport, nous commencerons par une présentation générale du jeu, en exposant les objectifs du joueur et les scènes du jeu. Ensuite, nous détaillerons la conception des blocs et entités qui composent le monde et leur gestion dynamique. Par la suite, nous aborderons la génération procédurale, en expliquant la méthode utilisée ainsi que les détails d'implémentation. Nous discuterons ensuite des contrôles du personnage et des différentes mécaniques de jeu, telles que le saut, le dash, etc. La partie suivante sera consacrée à la physique, notamment la gestion des collisions et des propriétés physiques comme la résistance de l'air. Enfin, nous reviendrons sur l'organisation du travail, la répartition des tâches et les outils utilisés pour mener à bien ce projet. Le rapport se conclura par un bilan du projet et les améliorations possibles.

## 2 Présentation du jeu

### 2.1 Présentation générale

### 2.2 Les objectifs du joueur

#### 2.2.1 Les écrous

#### 2.2.2 La sortie

### 2.3 Les scènes

## 3 Les blocks et entités du monde

### 3.1 La définition dynamique des blocks

### 3.2 Le stockage des blocks

## 4 La génération procédurale

## 5 Le personnage

### 5.1 Les contrôles de base

### 5.2 Le saut

#### 5.2.1 Le saut modulable

#### 5.2.2 Le double saut

### 5.3 Le dash

### 5.4 Le score et les vies

## 6 La physique

### 6.1 La gestion des collisions

#### 6.1.1 Les propriétés des blocks

### 6.2 Calcul de la résistance de l'air

## 7 Notre organisation

### 7.1 La définition des tâches

#### 7.1.1 La répartition du travail

### 7.2 Les outils utilisés

#### 7.2.1 Discord

#### 7.2.2 Google Docs

#### 7.2.3 Trello

#### 7.2.4 Git et Github

#### 7.2.5 IDE

## 8 Conclusion

### 8.1 Points d'amélioration

#### 8.1.1 Ce qu'on aurait voulu ajouter

#### 8.1.2 Wave Function Collapse

#### 8.1.3 Pathfinding amélioré

#### 8.1.4 La physique