

# RAPPORT DE PROJET

---

Développement d'un Jeu Vidéo 2D avec Godot  
Engine

"Escape the Maze"

**Étudiant :** Khalid ZEROUAL

**Projet :** Projet\_Unity-3d

**Année Académique :** 2025/2026

**Technologies :** Godot Engine, GitLab, ChatGPT, Itch.io

# Table des Matières

1. Introduction au Projet
2. Gestion de Version avec Git/GitLab
3. Apprentissage avec ChatGPT
4. Présentation du Jeu
  - Écran de Démarrage
  - Gameplay et Mécaniques
  - Éléments du Jeu
5. Déploiement sur Itch.io
6. Conclusion

## 1. Introduction au Projet

Ce projet consiste à développer un jeu vidéo 2D nommé **"Escape the Maze"** en utilisant le moteur de jeu **Godot Engine**. Le jeu est un labyrinthe d'action où le joueur doit naviguer à travers des obstacles, collecter des pièces et atteindre l'étoile pour terminer le niveau.

### Objectifs du Projet

- Apprendre à utiliser Godot Engine pour le développement de jeux 2D
- Mettre en pratique la gestion de version avec Git/GitLab
- Implémenter des mécaniques de jeu basiques (mouvement, collision, collectibles)
- Déployer le jeu sur une plateforme de distribution (Itch.io)

## 2. Gestion de Version avec Git/GitLab

### 2.1 Importance du Contrôle de Version

Pour ce projet, j'ai utilisé **GitLab** comme plateforme de gestion de code source. Le contrôle de version est essentiel dans le développement logiciel car il permet de :

- Suivre l'historique des modifications du projet
- Revenir à des versions antérieures en cas de problème
- Travailler sur différentes fonctionnalités en parallèle
- Documenter les changements apportés au code

### 2.2 Workflow Git Utilisé

Après chaque modification significative du projet, j'ai effectué les opérations suivantes :

```
# Ajout des fichiers modifiés git add . # Création d'un  
commit avec un message descriptif git commit -m  
"Description de la modification" # Envoi des modifications  
vers GitLab git push origin main
```

### 2.3 Exemples de Commits

- Initial commit : Création du projet Godot
- Ajout du personnage joueur : Implémentation du sprite et des contrôles
- Création du système de collision : Gestion des murs et obstacles
- Ajout des collectibles : Pièces et système de score
- Implémentation des ennemis : IA basique pour les slimes verts
- Création de l'écran de démarrage : Menu principal du jeu
- Finalisation et export HTML5 : Préparation pour le déploiement

## 3. Apprentissage avec ChatGPT

### 3.1 Utilisation de l'IA comme Outil d'Apprentissage

N'ayant jamais utilisé Godot auparavant, j'ai fait appel à **ChatGPT** comme assistant d'apprentissage. Cette approche m'a permis d'obtenir des réponses rapides et personnalisées à mes questions.

### 3.2 Exemples de Questions Posées

- **Bases de Godot** : "Comment créer une nouvelle scène dans Godot ?"
- **Mouvement du personnage** : "Comment implémenter les contrôles clavier pour déplacer un personnage ?"
- **Système de collision** : "Comment détecter les collisions entre le joueur et les obstacles ?"
- **Gestion du score** : "Comment créer un système de collecte de pièces avec affichage du score ?"
- **Export du jeu** : "Comment exporter un projet Godot en HTML5 pour le web ?"

### 3.3 Avantages de cette Méthode

- Apprentissage à mon propre rythme
- Explications adaptées à mon niveau
- Possibilité de poser des questions de suivi
- Accès à des exemples de code concrets

## 4. Présentation du Jeu "Escape the Maze"

### 4.1 Écran de Démarrage



Figure 1 : Écran de démarrage du jeu

#### Description de l'Écran de Démarrage

L'écran d'accueil présente le titre du jeu "**Escape the Maze!**" sur un fond gris épuré. Il affiche :

- **Titre principal** : Police grande et claire pour attirer l'attention
- **Instruction** : "Press <space>" pour indiquer au joueur comment démarrer
- **High Score** : Affichage du meilleur score (actuellement à 0)

Cet écran simple et efficace permet au joueur de comprendre immédiatement comment commencer à jouer. Le design minimaliste met l'accent sur la jouabilité plutôt que sur des éléments visuels complexes.

## 4.2 Vue du Gameplay



Figure 2 : Vue du niveau de jeu en cours

### Analyse du Gameplay

Cette capture d'écran montre le jeu en action avec tous ses éléments principaux :

#### Le Personnage Joueur

- Représenté par un petit personnage avec une casquette verte en bas à droite de l'écran
- Contrôlable avec les flèches du clavier ou WASD
- Peut se déplacer dans toutes les directions sur les surfaces accessibles

#### Les Obstacles

- **Murs rouges** : Obstacles infranchissables qui délimitent le labyrinthe
- Créent des corridors et des zones à explorer

- Le joueur doit naviguer intelligemment pour éviter les impasses



## Les Ennemis

- **Slimes verts** : Ennemis qui patrouillent dans le niveau
- Représentent un danger pour le joueur
- Le contact avec un ennemi peut réduire la vie du joueur



## Les Collectibles

- **Pièces dorées** : Dispersées dans tout le niveau
- Chaque pièce collectée augmente le score
- Encourage l'exploration complète du niveau



## Objets Spéciaux

- **Clé rose** : Visible près d'une porte verrouillée
- Nécessaire pour débloquer certains passages
- **Boîtes colorées** : Caisses qui peuvent être poussées ou qui contiennent des bonus



## Objectif Final

- **Étoile jaune** : Située en haut à droite
- Représente la sortie du niveau
- Atteindre l'étoile complète le niveau



## Design du Niveau

- **Zones bleues** : Chemins praticables avec motif en damier
- **Zones marron** : Autre type de terrain accessible
- Les différentes textures aident à la navigation visuelle

### 4.3 Détails du Niveau Avancé



Figure 3 : Vue d'une section différente du niveau

#### Évolution et Progression

Cette troisième capture montre le joueur ayant progressé dans le niveau. On observe plusieurs changements :

##### Progression du Joueur

- Le personnage s'est déplacé vers le centre du niveau
- Certaines pièces ont été collectées (moins de pièces visibles)
- Le joueur se rapproche de l'objectif final

##### Nouveaux Éléments

- **Clé rouge** : Apparaît dans la zone de gauche
- Probablement débloquée après avoir accompli une action spécifique
- Nécessaire pour accéder à de nouvelles zones

##### Variation de l'Environnement



- Mélange de différents types de terrains
- Placement stratégique des obstacles
- Chemins multiples vers l'objectif

### **Stratégie de Jeu**

Le joueur doit :

- Éviter les ennemis tout en collectant les pièces
- Trouver et utiliser les clés pour ouvrir les passages
- Planifier son itinéraire pour maximiser le score
- Atteindre l'étoile finale pour compléter le niveau

## **4.4 Mécaniques de Jeu Implémentées**

### **Système de Mouvement**

- Déplacement fluide dans 4 directions (haut, bas, gauche, droite)
- Vitesse de déplacement constante et adaptée
- Animation du personnage lors du mouvement

### **Système de Collision**

- Détection précise des collisions avec les murs
- Interaction avec les objets collectibles
- Gestion des contacts avec les ennemis

### **Système de Score**

- Points gagnés pour chaque pièce collectée
- Affichage en temps réel du score (visible en haut : "0")

- Sauvegarde du meilleur score (High Score)

### **Intelligence Artificielle Simple**

- Les slimes verts patrouillent dans des zones définies
- Mouvements prévisibles mais représentant un défi

## 5. Déploiement sur Itch.io

### 5.1 Pourquoi Itch.io ?

**Itch.io** est une plateforme de distribution de jeux indépendants qui offre plusieurs avantages :

- Hébergement gratuit et illimité
- Support natif des jeux HTML5
- Interface simple et accessible
- Communauté active de joueurs et développeurs
- Partage facile via URL directe

### 5.2 Processus d'Export HTML5

Pour déployer le jeu sur Itch.io, j'ai suivi ces étapes :

#### Étape 1 : Export depuis Godot

1. Ouverture du menu **Projet** → **Export**
2. Sélection du template **HTML5**
3. Configuration des paramètres d'export :
  - Résolution de l'écran
  - Nom du fichier HTML
  - Optimisation pour le web
4. Export du projet vers un dossier

#### Étape 2 : Préparation des Fichiers

L'export génère plusieurs fichiers nécessaires :

- `index.html` : Page principale du jeu
- `index.js` : Code JavaScript du jeu

- `index.wasm` : Fichier WebAssembly
- `index.pck` : Ressources du jeu

### Étape 3 : Compression des Fichiers

```
# Création d'une archive ZIP contenant tous les fichiers
zip -r escape-the-maze.zip index.html index.js index.wasm
index.pck
```

### Étape 4 : Upload sur Itch.io

1. Connexion sur [itch.io](https://itch.io)
2. Création d'un nouveau projet
3. Configuration du projet :
  - **Titre** : Escape the Maze
  - **Type** : HTML
  - **Description** : Jeu de labyrinthe d'action
4. Upload du fichier ZIP
5. Configuration de l'embed :
  - Activation de "This file will be played in the browser"
  - Définition de la résolution du canvas
6. Publication du jeu

## 5.3 Résultat Final

Le jeu est maintenant accessible en ligne via une URL Itch.io. Les joueurs peuvent :

- Jouer directement dans leur navigateur sans installation
- Partager le lien du jeu facilement

- Laisser des commentaires et des évaluations

**Compatibilité** : Le jeu fonctionne sur tous les navigateurs modernes supportant WebAssembly (Chrome, Firefox, Edge, Safari).

## 5.4 Optimisations Appliquées

- Compression des assets pour réduire la taille de téléchargement
- Optimisation des textures pour le web
- Configuration du chargement progressif
- Test sur différents navigateurs pour assurer la compatibilité

## 6. Conclusion

### 6.1 Compétences Acquisées

Ce projet m'a permis de développer plusieurs compétences importantes :

#### Compétences Techniques

- **Godot Engine** : Maîtrise des bases du développement de jeux 2D
- **GDScript** : Apprentissage du langage de programmation de Godot
- **Git/GitLab** : Gestion de version professionnelle
- **Déploiement Web** : Export HTML5 et hébergement sur Itch.io

#### Compétences Transversales

- **Apprentissage autonome** : Utilisation d'outils IA pour l'auto-formation
- **Résolution de problèmes** : Débogage et optimisation du code
- **Gestion de projet** : Organisation et planification des tâches
- **Game Design** : Conception de mécaniques de jeu engageantes

### 6.2 Défis Rencontrés

- **Courbe d'apprentissage de Godot** : Interface et concepts nouveaux à maîtriser
- **Système de collision** : Ajustements fins nécessaires pour un gameplay fluide
- **Optimisation des performances** : Maintenir un bon framerate avec plusieurs éléments
- **Export HTML5** : Configuration correcte pour le déploiement web

### 6.3 Améliorations Futures

Pour les prochaines versions, je prévois d'ajouter :

- **Plusieurs niveaux** : Progression avec difficulté croissante
- **Power-ups** : Objets donnant des capacités temporaires
- **Système de vies** : Gestion de la santé du joueur
- **Effets sonores et musique** : Ambiance audio immersive
- **Écran de Game Over** : Meilleur feedback en fin de partie
- **Leaderboard en ligne** : Compétition entre joueurs
- **Animations améliorées** : Mouvements plus fluides

## 6.4 Réflexion Personnelle

Ce projet a été une expérience d'apprentissage très enrichissante. L'utilisation de ChatGPT comme outil d'apprentissage s'est révélée particulièrement efficace, me permettant d'avancer rapidement tout en comprenant les concepts fondamentaux.

La gestion de version avec GitLab m'a appris l'importance de documenter ses modifications et de maintenir un historique propre du projet. Cette pratique sera essentielle dans ma future carrière de développeur.

Enfin, le déploiement sur Itch.io m'a montré qu'il est aujourd'hui facile de partager ses créations avec le monde entier, rendant le développement de jeux accessible à tous.

### Points Clés du Projet

- Jeu fonctionnel et jouable
- Gestion de version complète
- Déploiement web réussi
- Documentation détaillée
- Compétences transférables acquises