

Présentation

Jeux vidéo

Avec: Antoine, Adam, Amin.



Introduction

Nous serons un groupe de 4 personnes à travailler sur le projet de seconde notre projet consistera à créer un jeu de combat sur plateforme en 2D. Nous allons utiliser le moteur de jeu Godot et le langage de programmation GDScript pour créer le jeu.



Caractéristiques principales:

Le jeu sera inspiré de la licence Super Smash Bros, avec plusieurs personnages de licences différentes. Les joueurs pourront sélectionner différentes cartes pour combattre dans des environnements variés. Nous prévoyons également d'ajouter un mode multijoueur, où les joueurs pourront s'affronter sur le même ordinateur ou sur différents ordinateurs. Le jeu sera compatible avec les manettes et les claviers/souris pour offrir une expérience de jeu plus immersive.

```
1  extends Control
2
3  @export_range(0, 100, 1
4
5  var _backing: int = 0
6  var property:
7  > get:
8  >     return _backing
9  > set(value):
10 >     _backing = value
11
12 > func _ready():
13 >     super()
14 >     await $Button.button_down
15 >     $Label.text = "After"
16 >     $Label.text = await
17
18 > func coroutine():
19 >     await $Button.button_up
20 >     return "After second
21
```

• • •



Personnage/Cartes

Le jeu comprendra plusieurs personnages de différentes licences, comme des personnages de jeux vidéo ou de films. Nous allons travailler ensemble pour créer les personnages et les animations qui les accompagnent.

Le jeu proposera différentes cartes pour combattre dans des environnements variés, comme des châteaux ou des villes. Nous allons travailler ensemble pour créer les cartes et ajouter des éléments interactifs.

Compatibilité/ Multijoueur

Le jeu proposera un mode multijoueur où les joueurs pourront s'affronter sur le même ordinateur ou sur différents ordinateurs. Nous allons travailler ensemble pour implémenter cette fonctionnalité en mode multijoueur.

Le jeu sera compatible avec les manettes et les claviers/souris. Nous allons travailler ensemble pour assurer la compatibilité avec différents types de manettes et offrir une expérience de jeu fluide pour les joueurs.

Steam FA



Logiciel d'automatisation en python

Moi et mon équipe avons pris la décision de créer un logiciel d'automatisation de connexion pour Steam en utilisant Python. Cette idée est née de notre passion commune pour les jeux vidéo et notre désir d'améliorer l'expérience des utilisateurs de la plateforme Steam.

Le logiciel que nous développons sera capable de simplifier le processus de connexion à Steam en automatisant les étapes nécessaires. Il permettra aux joueurs d'économiser du temps et des efforts en évitant de saisir manuellement leurs identifiants de connexion à chaque fois qu'ils souhaitent accéder à leur compte Steam.

Notre équipe est consciente de l'importance de la sécurité des données personnelles et nous veillerons à ce que le logiciel soit sécurisé et fiable. Ensuite nous mettrons en place des mesures de protection telles que le chiffrement des informations de connexion pour garantir la confidentialité des utilisateurs.



Nous sommes convaincus que ce logiciel d'automatisation de connexion pour Steam sera très utile aux joueurs du monde entier. Il simplifiera leur expérience de jeu en leur permettant de se connecter rapidement et facilement à leur compte Steam. Notre objectif est de fournir un outil fiable et pratique qui améliorera la vie des utilisateurs de Steam.



Steam FA



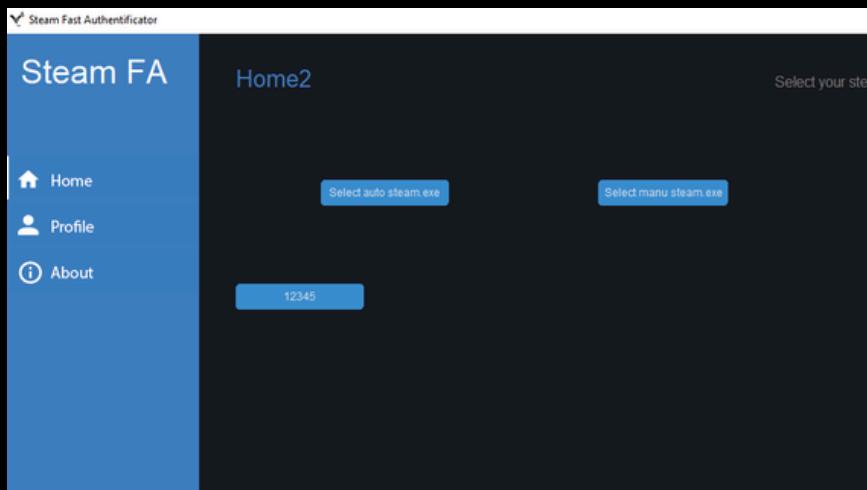
Présentation du logiciel

Le logiciel d'automatisation de connexion que nous développons pour Steam est construit en utilisant le langage de programmation Python, avec l'aide de la bibliothèque Tkinter pour créer l'interface utilisateur. Python est un choix idéal pour ce projet car il offre une grande flexibilité et une syntaxe claire.

Pour assurer la sécurité des données personnelles, nous avons décidé de stocker les informations de connexion de manière persistante en utilisant le format JSON. JSON est un format de données léger et largement utilisé, ce qui le rend pratique pour stocker les informations d'utilisateur. Cependant, pour protéger ces données sensibles, nous les chiffrons à l'aide d'une clé de chiffrement.

Le chiffrement des données nous permet d'ajouter une couche supplémentaire de sécurité. La clé de chiffrement est une séquence de caractères qui est utilisée pour transformer les données en un format illisible, sauf pour ceux qui possèdent la clé. Ainsi, même si les données sont compromises, elles ne peuvent être déchiffrées sans la clé appropriée.

```
decryption_key.key
1 a90jDn4Dhc_TnHm6HNuv[REDACTED]
```



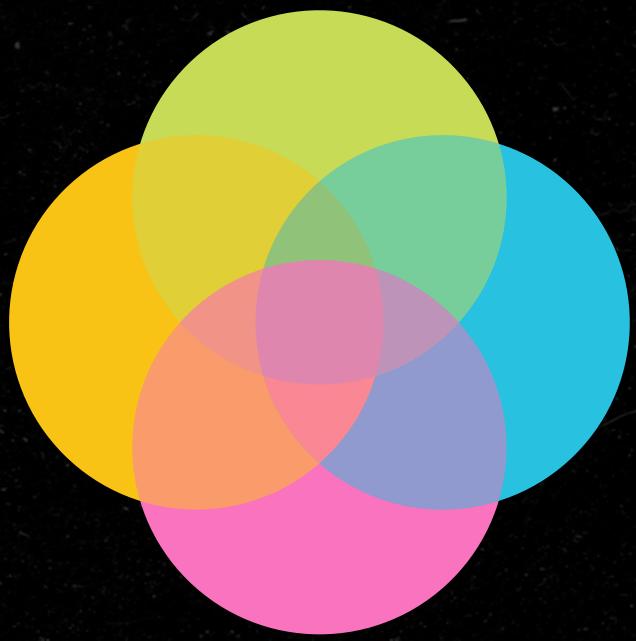
Steam FA



Figma

Pour le design de l'application, nous avons utilisé Figma, un site dédié pour les designer UX/UI.

Figma nous a aidé sur la conception design de notre logiciel, et nous a permis de se mettre d'accord sur les choix de styles de notre logiciel, tel que la charte graphique ou autres



Assets (éléments comme les boutons ou autres)

The screenshot shows the Figma application interface. On the left, there's a file tree with items like 'Home', 'Coup', 'Vector', 'Profile', and another 'Profile'. A white arrow points from the 'Assets' heading to the 'Vector' item in the tree. In the center, there are two wireframe designs: one for the 'Home' screen titled 'Select your steam.exe' and another for the 'Profile' screen titled 'Create your profile'. Both screens have input fields and buttons. On the right, there's a sidebar with tabs for 'Design', 'Prototype', and 'Inspect'. Under 'Design', there are sections for 'Background' (with a color hex code '1E1E1E' and a '100%' scale), 'Local styles', and 'Color styles'. A white arrow points from the 'Design' heading to the 'Color styles' section, specifically highlighting a blue circular icon labeled 'Boutton steam'.

Design (Styliser les éléments)

Steam FA

• • •

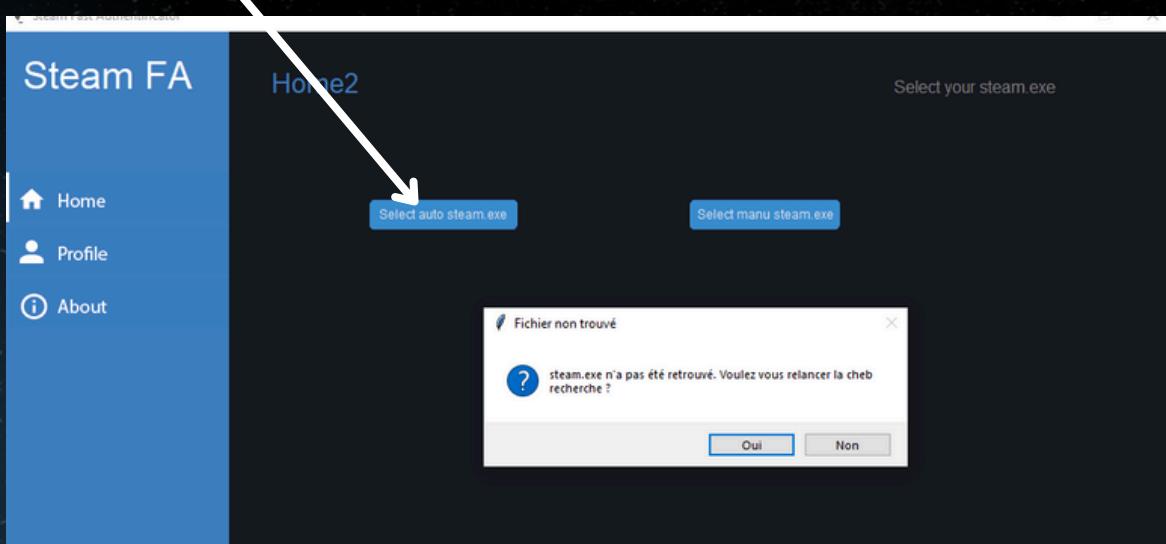
Fonctionnement du logiciel

Voici quelques images pour vous montrer comment le logiciel fonctionne.

Page d'accueil

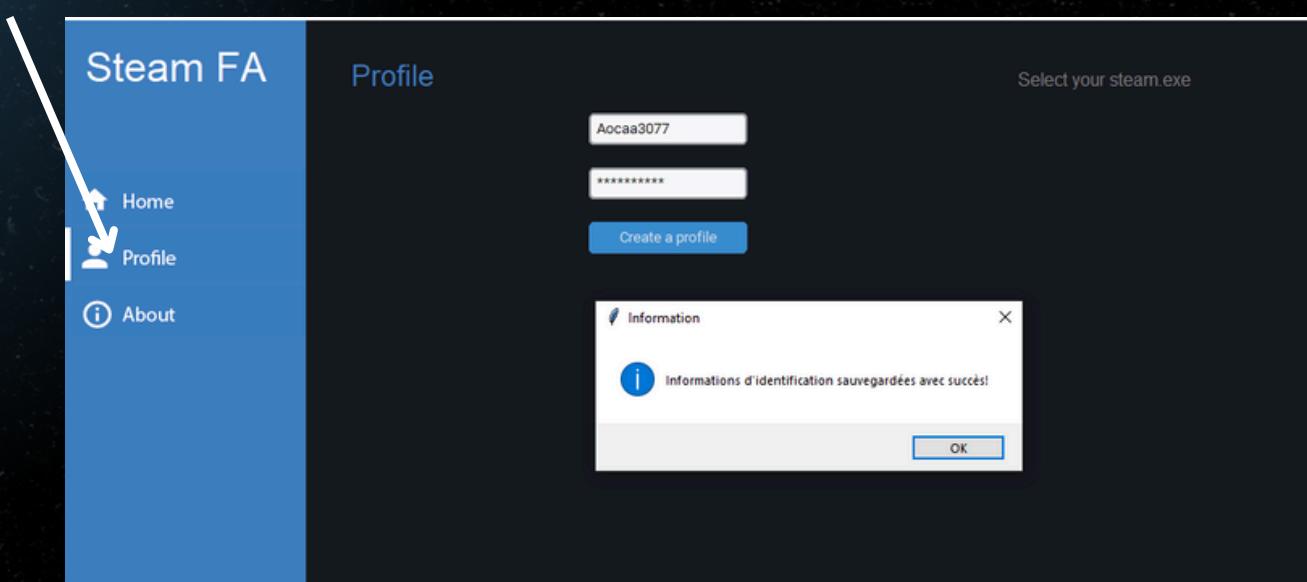
1er étape :

On appuie sur ce bouton qui va trouver automatiquement le fichier "steam.exe"

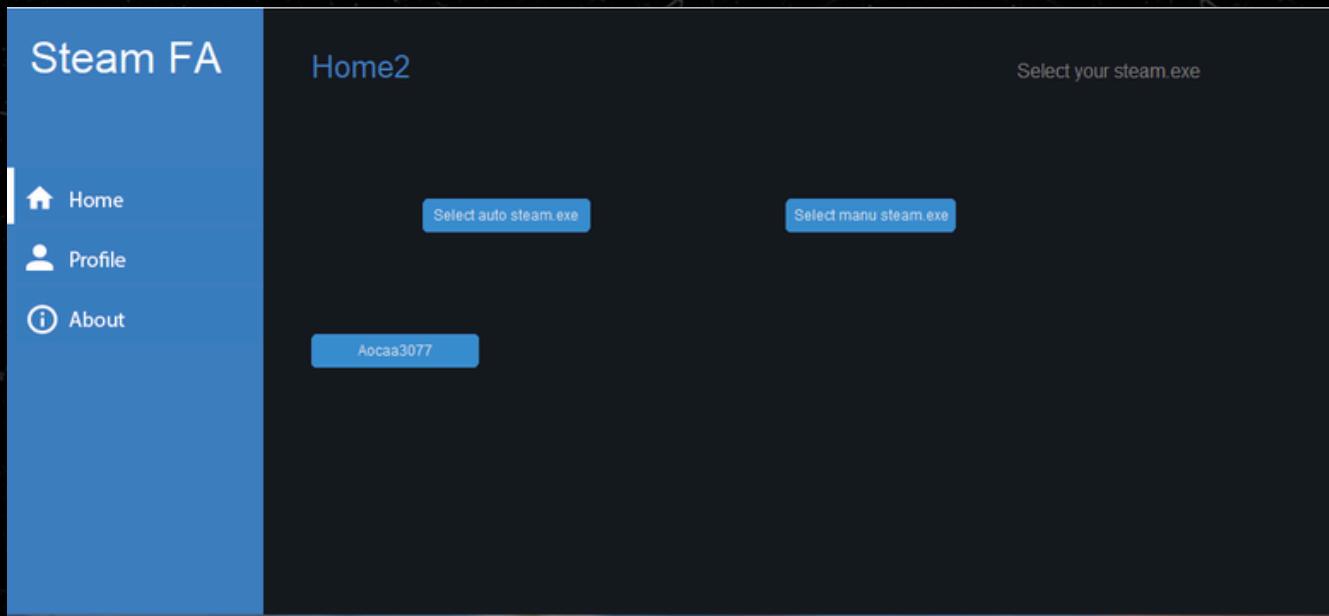


2ème étape :

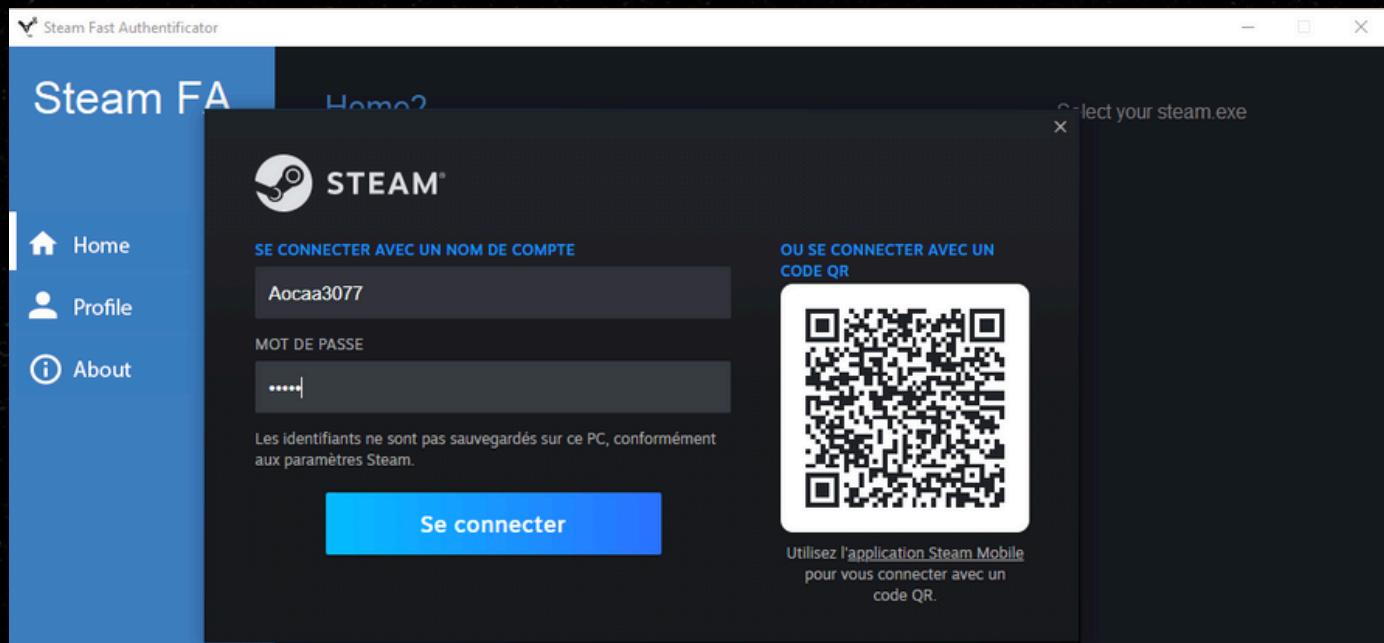
On crée un profile en mettant le nom d'utilisateur et le mot de passe



3ème étape : On appuie sur le profile qu'on vient de créer

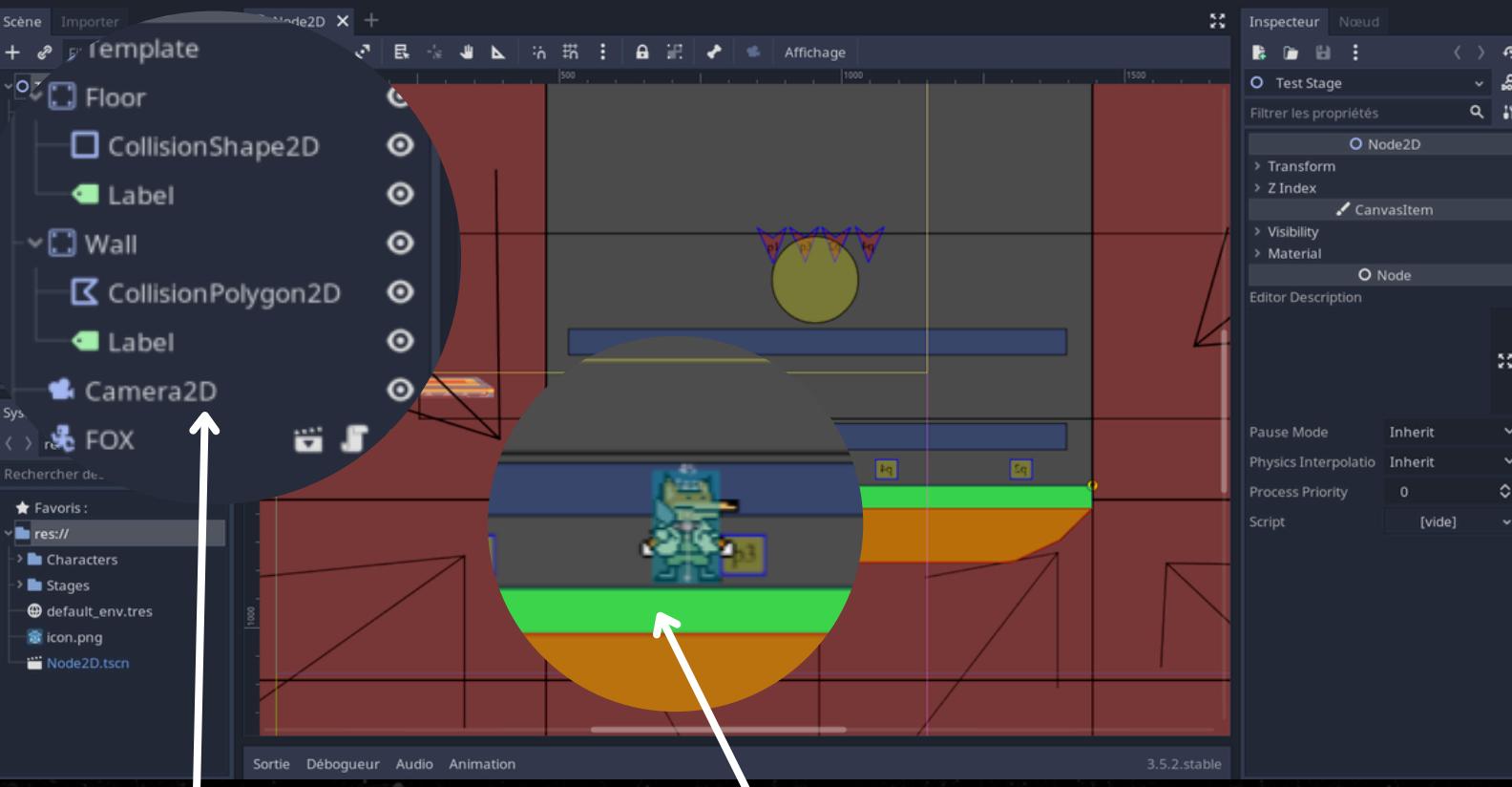


4ème étape :
L'application steam s'ouvrira automatiquement, et mettra les informations automatiquement aussi, puis se connectera.



Voici le lien pour télécharger le logiciel

https://drive.google.com/file/d/15BSyfDkajA9crXGKMfx98Cil9s7EqL6L/view?usp=drive_link



Structure du Jeux

Personnage Animé

Script Du Jeux

```
1 extends KinematicBody2D
2
3 #Globals Variables
4 var frame = 0
5
6 #Ground Variables
7 var velocity = Vector2(0,0)
8 var acc = 390
9 var walkSpeed = 200
10 var gravity = 1800
11 var jumpForce = 500
12 var maxJumpForce = 800
13 var doubleJumpForce = 1000
14 var maxAirspeed = 300
15 var airAccel = 25
16 var fallSpeed = 60
17 var fallingSpeed = 900
18 var maxFallSpeed = 900
19 var traction = 40
20 var rollDistance = 350
21 var dodgeSpeed = 500
22
23 func updateframes(delta):
24     frame += 1
25
26 func turn(direction):
27     var dir = 0
28     if direction:
29         dir = -1
30     else:
31         dir = 1
32     $Sprite.set_flip_h(direction)
```

Les Leçons Apprises : L'Échec d'un Projet de Jeu Vidéo Ambitieux

Introduction : Lancer un projet de jeu vidéo peut être une expérience passionnante, surtout lorsque l'on est lycéen et animé par une soif d'exploration. Cependant, l'enthousiasme peut parfois conduire à des ambitions démesurées et à une sous-estimation de l'expérience requise pour mener à bien un tel projet. C'est précisément cette combinaison d'ambitions démesurées et de peu d'expérience qui a conduit à l'échec de notre projet de jeu vidéo.

I. Des ambitions démesurées : Lorsque nous avons entrepris ce projet de jeu vidéo en tant que lycéens passionnés, nous avons immédiatement rêvé de créer un jeu révolutionnaire et ambitieux. Nous avons imaginé un vaste monde ouvert, des graphismes époustouflants, un scénario complexe et des mécaniques de jeu novatrices. Cependant, nous avons rapidement réalisé que nos ambitions dépassaient largement nos capacités actuelles.

II. Complications lors du développement : Les complications techniques ont commencé à émerger dès les premières étapes de développement. Les problèmes de code se sont accumulés, rendant la progression difficile. Les fonctionnalités que nous avions envisagées se sont révélées beaucoup plus complexes à implémenter que nous ne l'avions anticipé. Les bugs et les erreurs ont ralenti notre avancement, nous obligeant à passer beaucoup de temps à résoudre des problèmes plutôt qu'à ajouter des fonctionnalités au jeu.

Voici le lien pour installer le jeux

Touche du jeux pour se déplacer (ZQSD)

https://drive.google.com/file/d/1oQwACRJerEhYLttYvllPpKr0BOqmTQHw/view?usp=drive_link