Rapport de Soutenance – Labyrinouz

# Introduction

Labyrinouz est un projet ambitieux né de la volonté de combiner algorithmes, programmation système et design d’interface dans un environnement stimulant et ludique. Notre objectif était clair : créer une bibliothèque Rust capable de générer et résoudre des labyrinthes de manière modulaire, tout en proposant une interface utilisateur rétro, un site web interactif, et des fonctionnalités spécifiques à la génération procédurale.

# Équipe projet

- Ulysse T. : Chef de projet – Développement Rust  
- Jimmy N. : Design – Interface utilisateur  
- Mehdi T. : Algorithmes – Développement Web  
- Wali B. : Génération procédurale – IA

# Contexte et démarrage

Le projet a débuté à mi-novembre, avec une première phase de réflexion autour de l’objectif technique (génération/solution de labyrinthes), de l’esthétique (interface inspirée des OS rétro), et de la structure modulaire du code. Après avoir rédigé un cahier des charges fonctionnel, nous avons défini une répartition claire des tâches, établi une organisation Git, et commencé le développement parallèle des différents modules.

# Aspects techniques développés

• Génération de labyrinthes (Wali)  
 - Algorithmes procéduraux (DFS, Prim, Kruskal…)  
 - Modularité pour différents niveaux de complexité  
 - Intégration d'une IA de détection/chemin  
• Résolution de labyrinthes (Mehdi)  
 - Implémentation en Rust d’algorithmes comme BFS et A\*  
 - Visualisation interactive sur le site web  
 - Intégration front-end des résultats algorithmiques  
• Développement Web (Mehdi)  
 - Mise en page HTML/CSS de l’interface utilisateur  
 - Gestion des fenêtres interactives, animations et transitions  
 - Intégration des sections About, Tutorial, Chronology, Download  
• Interface utilisateur (Jimmy)  
 - Interface type Windows 98 avec effet pixel art  
 - Interaction entre les fenêtres (drag, z-index, ouverture dynamique)  
 - Navigation visuelle simple et fluide  
• Intégration Rust / Web (Ulysse)  
 - Création de la bibliothèque Rust (génération + solution)  
 - Tests unitaires et documentation  
 - Intégration Web via WebAssembly et structure modulaire

# Réalisations clés

- Bibliothèque Rust fonctionnelle (génération et résolution)  
- Interface Web rétro complète (HTML/CSS/JS + drag & drop)  
- Début d'intégration d'une IA de déplacement  
- Architecture modulaire claire  
- Site web documenté et interactif

# Conclusion

Ce projet nous a permis de combiner nos compétences en développement bas niveau, design, algorithmique et travail en équipe. Labyrinouz n’est pas seulement un outil, mais une expérience interactive que nous avons façonnée de A à Z. Ces deux mois nous ont appris à gérer un projet complexe en autonomie, et à faire dialoguer des univers très différents, de la logique pure au design visuel.