

# Programmation sûre et efficace

## Projet RUST : Développement d'un jeu

Solet D.

ESEO Angers

2025

# Cahier des charges

L'objectif est de développer un jeu de type *Turned-Based RPG* :

- Un personnage principal se déplace sur une carte où il peut rencontrer d'autres personnages / monstres à combattre.
- Les combats se font au tour par tour.

Le jeu devra s'exécuter dans le terminal :

- La carte sera donc limitée à une grille (ou plusieurs) avec possiblement des obstacles.
- Le côté RPG sera donc très limité au niveau de l'histoire (ce qui n'empêche pas de donner un contexte au démarrage du jeu ou dans un fichier séparé).

# Quelques idées

- Visibilité des « ennemies » :
  - Déplacement sur la grille → stratégie d'évitement.
  - Apparition aléatoire → système de fuite.
- Récupération de la vie :
  - Objet à trouver sur la carte ou à récupérer lors d'une victoire.
  - Emplacement particulier permettant de récupérer de la vie.
- Amélioration du personnage :
  - Niveau avec expérience et/ou objet de type *super bonbon*
  - Système d'armes améliorant les statistiques.
- Objectif du jeu :
  - Objectif précis : atteindre le 10ème étage, trouver le one piece, ...
    - Mécanisme de sauvegarde.
  - Jeux de scoring : pas de fin mais un score à battre
    - Sauvegarde du score.

# Contraintes techniques

- Concurrency (thread / mutex ou canaux de communication)
- Modularité
- Pas de warning (sauf justifié)
- Gestion des erreurs / options : pas de `unwrap`

# Évaluation

→ 33% du module.

## Barème

- Contraintes techniques (/10)
- Complexité du jeu (/3)
- Créativité (/3)
- Rapport (/4)

## À rendre

- Une notice avec l'histoire du jeu et les sources du code
- Un fichier ReadMe ou un mini-rapport :
  - Explication des différents modules
  - Indication de la répartition du travail
  - Justification des warnings (si nécessaire)
  - Documenter les *nouvelles* fonctionnalités du RUST ou les bibliothèques externes.