

# Projet UF - RogueLike

Le jeu sera un roguelike ou le but du jeu sera de tuer le dernier boss situé au 15ème étage. On démarre le jeu au premier étage, chaque étage est ainsi généré procéduralement. Un étage peut contenir plusieurs choses:

- Un coffre
- Une énigme
- Des monstres
- Un mini-boss
- Des bonus/malus
- etc..

Le joueur pourra ainsi au cours des différents étages débloquer des bonus pour l'aider à accéder au dernier étage.

Rôles de chacun:

- Sea Phanh Calvin: Développeur

Description de la stack technique :

- C#
- Firebase + redis

Designs patterns :

- Singleton -> GameManager pour m'assurer de n'avoir qu'un seul GameManager sinon je pourrais me retrouver avec plusieurs GameManager et ce n'est pas ce que je recherche

Le projet aura comme fonctionnalités:

- Une Architecture en services (IHM + API)
- Utiliser 3 design pattern utile
- Base de données

- Utilisation d'un ORM
- Génération d'un niveau aléatoire (qui reste logique)
- IA des monstres
- Utilisation d'objet lié au joueur
- Interaction avec l'environnement
- Choix de difficulté
- Parties personnalisés, choix multiples 3 de bonus ou malus
- Ajout d'une seed par nouvelle run

## **Fonctionnalités**

- Modèle de données : *Difficulté : 4*
  - Objets
    - un nom
    - une description
    - un effet
  - Niveaux
    - level
    - monstres
    - pièges
    - énigmes
    - mini-boss
    - boss
  - Armes
    - un nom
    - une portée
    - un pouvoir
  - Joueur
    - un pseudo
    - Points de vie
    - points d'action
    - dégats

- Monstre
  - un nom
  - points de vie
  - pouvoir
- Boss
  - un nom
  - points de vie
  - pouvoir
  - malus
- Seed
  - code
  - duration
- Une architecture en services (API + IHM): *Difficulté : 4*
- Une BDD nosql avec redis *Difficulté : 2*
- L'utilisation d'un ORM *Difficulté : 4*
- Génération d'un niveau aléatoire, avec des énigmes ou des monstres gérer en fonction de l'étage à laquelle se trouve le joueur *Difficulté : 10*
- IA des monstres *Difficulté : 12*
- Utilisation d'objet lié au joueur *Difficulté : 2*
- Interaction avec l'environnement *Difficulté : 4*
- Choix de difficulté (Système de Parangon) *Difficulté : 2*
- Partie Personnalisée, choix multiples 3 de bonus ou malus *Difficulté : 2*
- Ajout d'une seed par nouvelle run *Difficulté : 2*

*Degré de difficulté total : 48 points*

Planification:

Réalisation en premier du modèle de données avec chaque attribut.

Réaliser un Menu basique (Jouer, options, quitter)

Réaliser une première ébauche du joueur avec un systeme d'attaque, utilisation d'objet, ainsi que perte de vie et défaite.

Réaliser un niveau de testing, avec des interactions avec le joueur.

Faire une première ébauche de l'IA.

Réaliser le passage inter-niveaux.

Génération basique de niveaux avec que des monstres  
Faire la connexion à la BDD avec Redis, plus création de la seed.  
Réalisation de l'IA plus avancée pour les mini-boss et boss.  
Génération de niveaux avec mini-boss et boss  
Génération de niveaux avec des murs.  
Génération de niveaux avec pieges.  
Génération de niveaux avec enigmes.  
Ajout de compétences pour les mini-boss et boss  
Création des différents niveaux de parangons.  
Ajout des malus et bonus.  
Refaire le Menu avec d'autres fonctionnalités (choix de difficulté, partie personnalisé, mettre une seed)  
Faire un système de classement.