

Jeu sur iOS

Créer un jeu complet sur iOS

I. Contexte et objectif.....	1
II. Stack technique.....	1
III. Résultat.....	1
IV. Conclusion.....	3

I. Contexte et objectif

Contexte: en août 2024, je me suis lancé un défi personnel : développer un jeu complet pour iOS. L'objectif n'était pas seulement de créer un gameplay, mais de prendre en charge l'ensemble des aspects liés au développement d'un jeu mobile, de la conception à la publication. Cela comprend notamment la gestion de la base de données, la mise en place de mécanismes anti-triche, l'optimisation des performances, et bien d'autres éléments souvent sous-estimés en début de projet.

Ce projet est toujours en cours, et j'espère le publier un jour sur l'App Store. Il représente pour moi un terrain d'expérimentation ambitieux et extrêmement enrichissant.

Objectif: mener un projet long terme sur une création originale, tout en apprenant à créer un projet utilisant plusieurs technologies.

II. Stack technique

Pour le développement de cette application iOS, le choix de Swift s'est naturellement imposé. J'utilise plusieurs frameworks modernes de l'écosystème Apple :

- SpriteKit pour la gestion des graphismes et de l'animation du jeu
- SwiftUI pour la construction de l'interface utilisateur
- SwiftData pour la persistance locale des données

Côté serveur, j'ai opté pour une architecture en Node.js, à laquelle s'ajoute une petite partie en Java, utilisée pour la création de certains objets ce qui, à mon goût, est plus simple en Java.

La base de données repose sur PostgreSQL, robuste et bien adaptée à la structure relationnelle des données de jeu.

III. Résultat

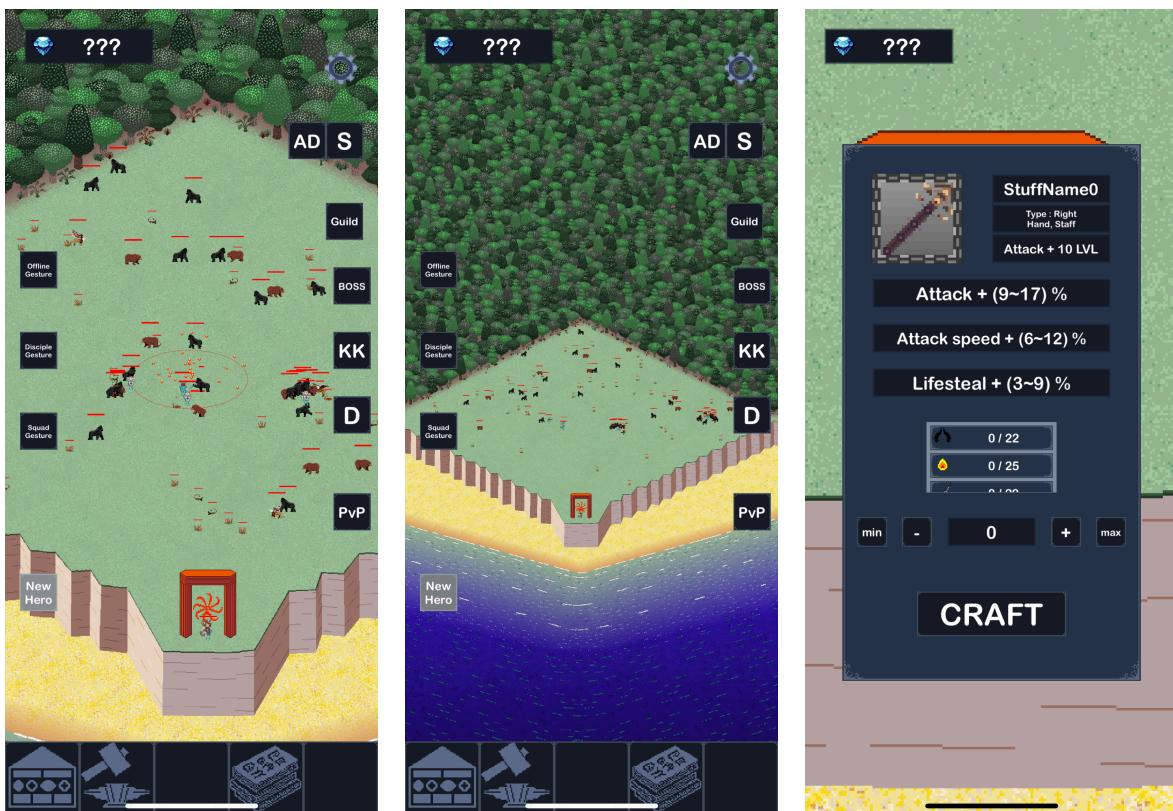
Dans ce jeu, le joueur incarne le gestionnaire d'une troupe de héros. Ces héros combattent automatiquement des monstres et des boss pour gagner en puissance. Le rôle du joueur n'est pas de les contrôler en temps réel, mais de prendre les bonnes décisions stratégiques pour les héros. Le joueur doit :

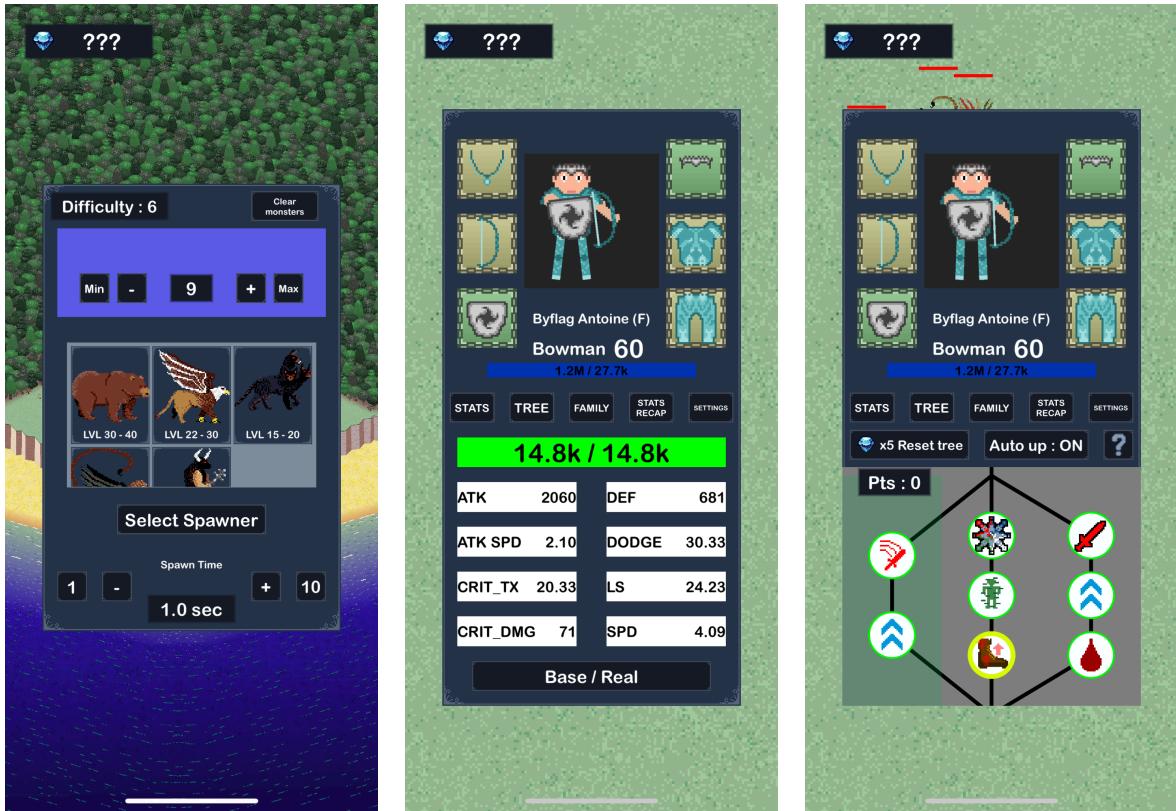
- choisir quel équipement leur attribuer
- définir quels sorts ils doivent apprendre
- composer les équipes
- gérer la progression via un système de lignées

Le jeu intègre également plusieurs fonctionnalités multijoueurs, comme :

- un mode arène, où les meilleurs héros de deux joueurs s'affrontent automatiquement, avec un système de classement compétitif
- un système de guildes, encore en cours de développement, mais qui pose déjà les bases pour des mécaniques communautaires à venir

Voici quelques screens.





IV. Conclusion

Travailler sur un projet aussi long est une expérience que j'apprécie particulièrement. Ce que je préfère, c'est sans doute la réflexion autour de l'architecture logicielle. Concevoir une structure claire, évolutive et réutilisable exige de se projeter dans le futur du projet, d'anticiper les besoins à venir, et d'optimiser dès maintenant ce qui pourrait devenir bloquant plus tard. Même si le jeu n'est pas encore très gourmand, les performances restent un sujet important car chaque optimisation mise en place aujourd'hui est un gain pour demain.

Ce projet m'a aussi permis de renforcer mes compétences en intégration multi-technologies. Relier Swift, Node.js, Java et PostgreSQL tout en maintenant une logique cohérente demande rigueur et organisation.

Il reste encore beaucoup de travail à faire : interface utilisateur, expérience de jeu, équilibrage, nouvelles fonctionnalités... mais c'est justement cette richesse et cette complexité qui rendent le projet aussi formateur et stimulant.