

1. Introduction

Dans le cadre de notre projet de groupe de Technologie Objet, notre équipe GH4 a développé un simulateur d'écosystème, un outil informatique permettant de modéliser les interactions au sein d'un environnement naturel. Ce simulateur vise à reproduire de manière réaliste les dynamiques au sein de l'environnement, incluant la croissance des populations, les relations prédateur-proie, ainsi que les impacts des changements environnementaux.

L'objectif principal de ce projet est de fournir une plateforme éducative et scientifique pour étudier les mécanismes écologiques et tester diverses hypothèses sur la stabilité et la résilience des écosystèmes. Les utilisateurs peuvent modifier certains paramètres directement sur l'interface du simulateur, ce qui permet d'explorer les processus écologiques et d'anticiper les effets des perturbations environnementales.

2. Fonctionnalités Principales

Notre simulateur d'écosystème intègre plusieurs fonctionnalités clés qui permettent de modéliser et d'analyser les interactions au sein de l'environnement simulé. Voici un aperçu de ces fonctionnalités :

1. Interactions entre espèces (Réalisé à l'itération 3)

La fonctionnalité fondamentale du projet réside dans les interactions entre les différentes espèces. Les individus peuvent tuer leurs proies s'ils en trouvent à portée, et peuvent se reproduire si les conditions sont favorables (suffisamment de nourriture et la présence d'un individu du sexe opposé).

2. Comportement ajusté à l'environnement (Réalisé à l'itération 3)

Les individus réagissent en fonction de leur environnement. Cela inclut la création de chemins vers les proies ou vers d'autres individus de la même espèce pour la reproduction, ce qui permet de simuler des comportements réalistes.

3. Contrôle d'un individu (Réalisation à 20%)

Pour une immersion plus grande, l'utilisateur peut prendre le contrôle d'un individu. Actuellement, seul le renard est contrôlable, et il ne peut pas interagir avec les autres individus.

4. Visualisation des données (Réalisé à l'itération 3)

Pour offrir une vision claire et non biaisée des populations, nous avons implémenté un affichage en diagramme à barres dynamique. Ce graphique s'adapte en temps réel au nombre d'individus de chaque espèce, facilitant ainsi l'analyse des dynamiques de population.

5. Ajout d'individus à tout moment (Réalisé à 100%)

Une fonctionnalité permet à l'utilisateur d'ajouter un nombre choisi d'individus de n'importe quelle espèce à tout moment. Cela permet d'ajuster la balance entre les espèces et de tester différents scénarios.

Ces fonctionnalités forment le cœur du simulateur et permettent aux utilisateurs d'explorer diverses dynamiques écologiques, de tester des hypothèses et de visualiser les impacts des interactions et des changements environnementaux de manière interactive et éducative.

2. Présentation Technique

Architecture de l'Application

L'architecture de notre simulateur d'écosystème est conçue pour être modulaire et évolutive. Elle se décompose en plusieurs composants principaux, chacun responsable d'une partie spécifique du système. Voici une vue d'ensemble de l'architecture avec l'aide de diagrammes UML :

1. **Entités** : Contient les classes représentant les différentes entités de l'écosystème (ex: **Poule**, **Renard**, **Vipere**).
2. **Main** : Contient les classes principales et les classes de gestion de l'application (ex: **Main**, **Ecran**, **UI**).
3. **Objet** : Contient les classes représentant des objets génériques de l'écosystème (ex: **Eau**, **Plantes**).
4. **Terrain** : Contient les classes qui représentent et gèrent les différents types de terrains dans l'écosystème (ex: **GestionTerrain**, **Terrain**).

Diagramme de Classe

Le diagramme de classe ci-dessus illustre les relations entre les principales classes de notre simulateur. Les entités de l'écosystème héritent toutes d'une classe **Entite** de base, ce qui permet de gérer les comportements communs de manière polymorphe.

Choix de Conception

Nous avons fait plusieurs choix de conception pour optimiser notre simulateur :

1. Scénario de Base :

- Nous avons décidé de commencer avec un scénario simple incluant trois espèces : poules, renards et vipères. Cette simplicité initiale a permis de créer une base solide sur laquelle nous pourrions ajouter des fonctionnalités plus complexes ultérieurement.

2. Comportement des Individus :

- Nous avons cherché à simuler des relations prédateur-proie réalistes. Cependant, en raison des limitations de temps et de ressources, nous avons limité les comportements à des interactions de base comme la prédation et la reproduction.

3. **Affichage :**

- En raison de notre manque d'expérience avec les moteurs de jeu et des contraintes de temps, nous avons opté pour utiliser Swing pour l'affichage. Cette décision nous a permis de développer une interface 2D fonctionnelle sans trop de complexité.

Problèmes Rencontrés et Solutions

1. **Moteur de Jeu :**

- Nous avons initialement envisagé d'utiliser un moteur de jeu pour améliorer les graphismes. Cependant, nous avons rencontré des difficultés à intégrer ces moteurs dans notre projet. En conséquence, nous avons décidé de nous replier sur Swing, ce qui nous a permis de progresser rapidement sans compromis sur la fonctionnalité.

2. **Bugs :**

- Comme tout projet logiciel, nous avons rencontré plusieurs bugs. Par exemple, des problèmes de collision et de pathfinding ont été résolus en implémentant des algorithmes de gestion des collisions et de recherche de chemin plus robustes.

Notre approche technique a permis de créer un simulateur fonctionnel tout en restant flexible pour des améliorations futures. Les choix de conception et les solutions apportées aux problèmes rencontrés ont été guidés par l'objectif de fournir un outil éducatif et scientifique robuste et évolutif.

Organisation de l'Équipe

Notre équipe, baptisée GH4, s'est organisée de manière efficace pour mener à bien le développement du simulateur d'écosystèmes. Voici la répartition des rôles :

- **SCRUM Master** : Pascal Benjamin
- **Équipe de Développement** :

- Babouche Elias

- Chellal Najlaa
- Dubois Emmanuel
- Durollet Pierre
- Renoud-Grappin Youri
- Seye Papa Samba

Le rôle du SCRUM Master était de faciliter les réunions, de suivre l'avancement du projet et de s'assurer que l'équipe respectait les principes agiles.

Mise en Œuvre des Méthodes Agiles

Dès le début du projet, nous avons adopté une approche agile pour structurer notre travail. Cette méthodologie nous a permis de rester flexibles et réactifs aux besoins et aux défis rencontrés. Voici comment nous avons procédé :

1. Définition des Attentes et des Fonctionnalités :

- Nous avons commencé par établir une liste des fonctionnalités attendues et identifier les éléments de base qui pourraient être améliorés ou complexifiés par la suite.
- Un premier diagramme UML a été élaboré pour représenter la version la plus basique du projet.

2. Formation de Sous-Groupes :

- L'équipe a été divisée en binômes ou trinômes pour travailler simultanément sur différents aspects du projet, accélérant ainsi le développement.
- Cette division a permis une spécialisation des tâches et une meilleure gestion du temps et des compétences.

3. Développement Iteratif :

- Nous avons initialement développé le simulateur sans l'interface graphique pour nous concentrer sur la fonctionnalité de la simulation elle-même.
- Une fois la simulation validée, l'équipe s'est concentrée sur l'intégration et l'amélioration de l'interface graphique.

4. Réunions Périodiques :

- Des réunions courtes et régulières (stand-ups) ont été tenues pour discuter de l'avancement du projet, des obstacles rencontrés et des prochaines étapes.
- Ces réunions ont permis à chaque sous-groupe de rester synchronisé et de s'assurer que tous les membres étaient sur la même longueur d'onde.

5. **Utilisation d'Outils de Gestion de Projet :**

- **GitHub** : Utilisé pour le contrôle de version et la collaboration sur le code.
- **Trello** : Utilisé pour suivre les tâches à faire, en cours et terminées. Chaque carte sur Trello représentait une tâche ou une fonctionnalité, facilitant la visualisation de l'avancement du projet.

Cette organisation a été particulièrement bénéfique pour gérer efficacement le projet, d'autant plus qu'il ne s'agissait pas de notre seule activité pendant cette période. La méthodologie agile nous a aidés à rester focalisés et à adapter notre plan de travail en fonction des retours et des résultats obtenus à chaque étape.