

---

## Rapport 3

---

# Simulateur d'écosystèmes

Groupe GH4

Mars 2024 - Mai 2024

# Sommaire

1. Introduction
2. Fonctionnalités Principales
3. Découpage de l'Application
4. Diagramme de Classe
5. Choix de Conception
6. Organisation de l'Équipe

## 1 Introduction

Dans le cadre de notre projet de groupe de Technologie Objet, notre groupe 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, le tout en pouvant modifier certains paramètres directement sur l'interface du simulateur. Notre simulateur offre une modélisation simple afin d'explorer les processus écologiques et d'anticiper les effets des perturbations environnementales.

## 2 Fonctionnalités Principales

1. Interactions entre espèces
2. Comportement ajusté à l'environnement
3. Contrôle d'un individu
4. Visualisation des données
5. Ajout d'individus à tout moment

### **Interactions entre espèces** Réalisé itération 3

La fonctionnalité fondamentale du projet, que les individus interagissent entre eux, cela est simulé par des individus qui tuent leurs proies si il y en a à portée. Mais aussi qui peuvent procréer s'ils sont dans les bonnes conditions (suffisamment de nourriture et un individu du sexe opposé).

#### **Comportement ajusté à l'environnement** Réalisé itération 3

Afin de mieux simuler le comportement des individus, nous avons implémenté une réaction de la part d'un individu en fonction de son environnement. Ceci est caractérisé par la création du chemin vers les proies ou vers les individus de la même espèce afin de procréer.

#### **Contrôle d'un individu** Réalisation 20%

Pour se sentir plus impliqué et ne pas avoir qu'à attendre l'affichage des données, l'utilisateur peut prendre le contrôle d'un individu. Pour l'instant il n'est possible que de prendre l'apparence d'un renard qui ne peut pas interagir avec les autres individus.

#### **Visualisation des données** Réalisé itération 3

Afin d'avoir une vision générale et de ne pas être biaisé par nos yeux qui pourraient mal interpréter le nombre d'individus d'une espèce en regardant sur la carte, nous avons implémenter un affichage en diagramme à bars dynamique, qui s'adapte en temps réel au nombre d'individus.

#### **Ajout d'individus à tout moment** Réalisation 100%

Nous avons ajouté une fonctionnalité permettant à l'utilisateur d'ajouter un nombre de son choix d'individus de n'importe quelle espèce, ceci permet de pouvoir ajuster la balance entre les espèces.

### **3 Découpage de l'application**

1. Entités : Contient les classes représentant les différentes entités de l'écosystème.
2. Main : Contient les classes principales et les classes de gestion de l'application.
3. Object : Contient les classes représentant des objets génériques de l'écosystème.
4. Terrain : Contient des classes qui représentent et gèrent les différents types de terrains dans l'écosystème.

```
Projet Long
├── Entités
│   ├── Entite.java
│   ├── GestionEntite.java
│   ├── Joueur.java
│   ├── Node.java
│   ├── Poule.java
│   ├── Renard.java
│   └── Vipere.java
├── Main
│   ├── ActionClavier.java
│   ├── Ecran.java
│   ├── FonctionUtiles.java
│   ├── Main.java
│   ├── Son.java
│   ├── UI.java
│   └── VerifierCollision.java
├── Object
│   ├── Eau.java
│   ├── GererObject.java
│   ├── JeuObject.java
│   └── Plantes.java
└── Terrain
    ├── GestionTerrain.java
    └── Terrain.java
```

## 4 Diagramme de Classe

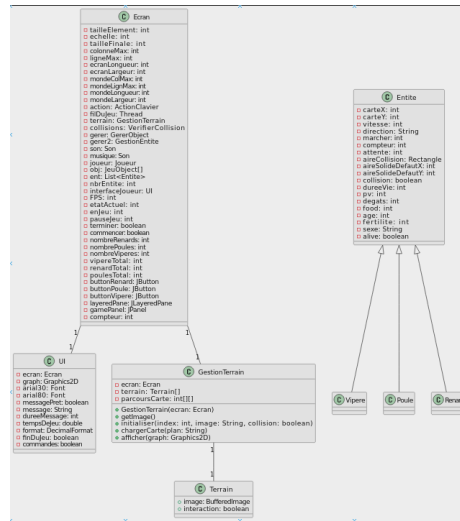


Figure 1: Enter Caption

## 5 Choix de Conception

## 5.1 Choix Réalisés

Nous avons fait différents choix, notamment sur l'importance des fonctionnalités. Ceux-ci sont:

1. Scénario de base
2. Comportement des individus
3. Affichage

**Scénario de base** Nous avons décidé de faire un scénario basique avec comme espèces : des poules, des renards et des vipères. Cette simplicité a permis de créer une base afin de potentiellement complexifier l'application si besoin, par exemple en ajoutant des comportements liés à de la nourriture, voire d'autres espèces.

**Comportement des individus** Notre groupe a décidé de tenter de rester proche de la réalité en simulant des rapports prédateurs-proies entre les 3 espèces. Le manque d'autre paramètres externes implémentés limite la possibilité d'ajouter des comportements.

**Affichage** Du fait du manque d'expérience sur des moteurs de jeu, et les limitations en termes de durée nous ont forcé à nous restreindre à Swing. Sur lequel, de part sa simplicité, nous as permis de rester sur une carte en 2 dimensions.

## 5.2 Problèmes rencontrés et Solutions :

1. Moteur de Jeu
2. Bugs

**Moteur de Jeu** Nous avons tout d'abord pensé utiliser un moteur de jeu afin d'avoir de meilleurs graphiques, cependant, nous n'avons pas réussi à en ajouter aux librairies du projet. Donc nous nous sommes rabattus sur Swing.

**Bugs** Comme dans tout jeu, il va y avoir des bugs, et nous n'avons pas été épargnés. Que cela soit en sortant du cadre normalement autorisé, n'arrivant pas à manger leurs proies ou la recherche de chemin vers les proies qui ne fonctionne pas si c'est trop loin.

## 6 Organisation de l'Équipe

**SCRUM Master** : PASCAL Benjamin

**Équipe de développement** : BABOUCHE Elias, CHELLAL Najlaa, DUBOIS Emmanuel, DUROLLET Pierre, RENOUD-GRAPPIN Youri, SEYE Papa Samba

### 6.1 Mise en Oeuvre des Méthodes Agiles

Au début, nous avons détaillé ce que nous attendions de ce projet, quelles fonctionnalités seraient attendues, trouver des fonctionnalités de base qui pourraient être améliorées, complexifiées.

De là nous avons développé un premier diagramme UML, qui ne représente que la version la plus basique du projet.

C'est à partir de là que l'équipe de développement a été séparée en binômes ou trinômes afin de travailler sur différents aspects en même temps et accélérer le développement.

Afin de vérifier l'avancement, nous avons fait le simulateur sans l'interface graphique. Le but primaire étant de voir si la simulation fonctionne correctement. Cela car c'est la partie la plus importante du projet, le visuel n'étant que secondaire par rapport à la simulation.

De courtes réunions périodiques ont permises de se mettre d'accord sur l'avancement du projet, ce qu'il reste à faire et afin de travailler avec notre binôme ou trinôme. L'utilisation de GitHub et de Trello nous ont permis, en gardant une trace, de travailler en collaboration et de suivre l'avancement du projet, de ce qui doit être fait, ce qui est en train d'être fait, et ce qu'il faut faire.

Cela a été particulièrement utile afin de ne pas être perdu dans les objectifs à faire, et ce du fait de la durée du projet ainsi que le fait que c'est pas la seule chose que nous avons à faire.