

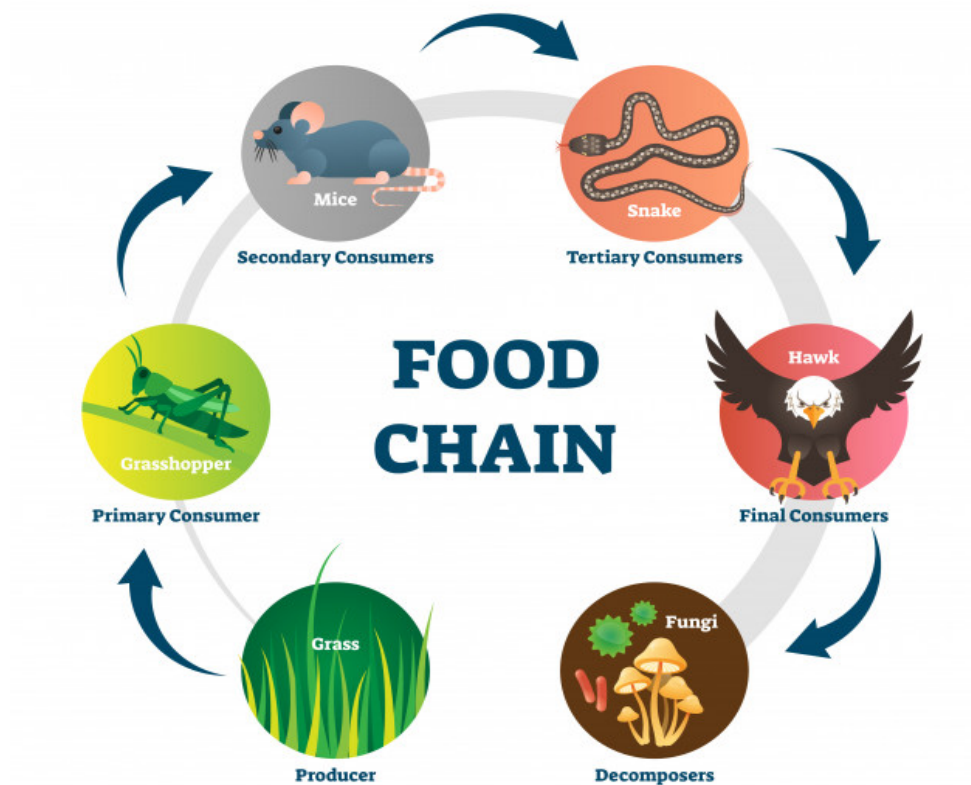
Université de Cergy-Pontoise

## RAPPORT

pour le projet Génie Logiciel  
Licence d'Informatique deuxième année

## Alimentaire

AIT MANSOUR YASSIN, BALET THIBAUT, BARIL ZACHARIE



Avril 2020

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Contexte et objectif du projet . . . . .	4
<b>2</b>	<b>Spécification du projet</b>	<b>5</b>
2.1	Notions de base et contraintes du projet . . . . .	5
2.1.1	Territoires et climats . . . . .	5
2.1.2	Espèces et chaînes alimentaires . . . . .	5
2.1.3	Plaines . . . . .	5
2.1.4	Marécages . . . . .	6
2.1.5	Glacial . . . . .	6
2.1.6	Savane . . . . .	7
2.1.7	Lac . . . . .	7
2.1.8	Décomposeurs . . . . .	8
2.1.9	Migrations . . . . .	8
2.2	Fonctionnalités attendues . . . . .	8
<b>3</b>	<b>Conception et réalisation du projet</b>	<b>9</b>
3.1	Architecture globale du logiciel . . . . .	9
3.2	Conception des classes de données . . . . .	10
3.3	Conception des traitements (processus) . . . . .	11
3.3.1	La gestion des espèces . . . . .	11
3.3.2	La mort et la transformation minérale . . . . .	11
3.3.3	Le mouvement des espèces . . . . .	11
3.4	Conception de l'IHM graphique . . . . .	12
3.4.1	Le menu de sélection d'écosystèmes . . . . .	12
3.4.2	La fenêtre de simulation d'un écosystème . . . . .	12
<b>4</b>	<b>Manuel utilisateur</b>	<b>14</b>
<b>5</b>	<b>Déroulement du projet</b>	<b>19</b>
5.1	Réalisation du projet par étapes . . . . .	19
5.2	Répartition des tâches entre membres de l'équipe . . . . .	19
<b>6</b>	<b>Conclusion et perspectives</b>	<b>21</b>
6.1	Résumé du travail réalisé . . . . .	21
6.2	Améliorations possibles du projet . . . . .	21

## Table des figures

1	Chaîne alimentaire des plaines . . . . .	5
2	Chaîne alimentaire des marécages . . . . .	6
3	Chaîne alimentaire de la banquise . . . . .	6
4	Chaîne alimentaire de la savanne . . . . .	7
5	Chaîne alimentaire du lac . . . . .	7
6	Structure du logiciel . . . . .	9
7	UML du squelette des classes de données . . . . .	10
8	Schéma de la fenêtre de sélection d'écosystèmes . . . . .	12
9	Schéma de la fenêtre de simulation . . . . .	13
10	Fenêtre de démarrage . . . . .	14
11	fenêtre de simulation . . . . .	15
12	Simulation de l'écosystème . . . . .	16
13	Zone de simulation . . . . .	17

14	Partie informations . . . . .	18
15	Diagramme de répartition des tâches . . . . .	20

## Liste des tableaux

1	Rangement des espèces de l'écosystème savane . . . . .	11
---	--	----

## Remerciements

Les auteurs du projet voudraient remercier vivement monsieur Tianxiao Liu, enseignant supérieur en catégorie A au Département des Sciences Informatiques de l'UFR Sciences et Techniques de l'Université de Cergy-Pontoise pour l'aide et le soutien apporté au groupe durant la réalisation de ce projet.

# 1 Introduction

Dans le cadre de notre seconde année du cycle de la License en Informatique, il nous est proposé un projet de 3 à 4 mois nous permettant de mettre en pratique nos connaissances en programmation et en la maîtrise du langage JAVA ayant pour finalité la conception et le développement d'une simulation d'un écosystème comme il nous a été demandé.

Dans cette section, nous allons présenter brièvement l'objectif du projet

## 1.1 Contexte et objectif du projet

Dans un écosystème, les liens qui unissent les espèces sont le plus souvent d'ordre alimentaire. On distingue trois catégories d'organismes :

1. Les producteurs (surtout les végétaux chlorophylliens, capables, grâce à la photosynthèse, de fabriquer de la matière organique à partir de dioxyde de carbone et de lumière solaire, mais aussi d'autres organismes autotrophes, certains étant à la base de chaînes alimentaires totalement indépendantes de l'énergie solaire).
2. Les consommateurs (les animaux), il existe trois types de consommateurs :
  - les herbivores qui se nourrissent des producteurs, on les appelle aussi consommateurs primaires,
  - les carnivores primaires, ou encore consommateurs secondaires, qui se nourrissent des herbivores,
  - les carnivores secondaires, appelés également consommateurs tertiaires, qui se nourrissent des carnivores primaires.
3. Les décomposeurs (les bactéries, champignons) qui dégradent les matières organiques de toutes les catégories et restituent au milieu les éléments minéraux.

Ces relations forment des séquences où chaque individu mange le précédent et est mangé par celui qui le suit, on parle de chaîne alimentaire. Chaque maillon est un niveau trophique. La niche écologique est ce que partagent deux espèces animales quand elles habitent le même milieu et qu'elles ont le même régime alimentaire. Ainsi, deux espèces ayant la même niche sont en "compétitions".

Reconsidérons à présent la croissance exponentielle. À l'évidence, il n'est pas réaliste d'imaginer qu'une population animale puisse croître exponentiellement sans rencontrer à un moment ou à un autre des limites à sa croissance. En effet, elle exploite des ressources qui sont évidemment limitées, ainsi en va-t-il de l'herbe pour nos lièvres, ou plus simplement encore de la superficie du territoire disponible.

## 2 Spécification du projet

### 2.1 Notions de base et contraintes du projet

#### 2.1.1 Territoires et climats

Nous avons décidé de créer 5 écosystèmes différents :

1. des plaines avec un climat tempéré ;
2. une zone humide/marécageuse ;
3. une savane avec un climat très chaud ;
4. une zone glaciale avec un climat très froid ;
5. une zone « aquatique » au centre.

Dans ces territoires vivront des espèces qui se mangeront entre-elles. Certaines espèces, en quête de survie, tenteront de migrer vers un autre territoire et de s'y adapter mais il faudra respecter une certaine logique, un reptile ne pourrait pas vivre dans les montagnes.

#### 2.1.2 Espèces et chaînes alimentaires

Le but principal étant la simulation d'une variété d'espèces, il y aura des chaînes alimentaires pour chaque territoire. Tous individus aura une durée de vie. Chaque espèce aura une fonctionnalité de reproduction qui se fera en fonction du nombre de ressources disponibles, du nombre de calories consommées.... Il se peut que certaines espèces vivant dans une même « zone » et ayant le même régime alimentaire finissent par entrer en conflit, or pour cohabiter il faut soit un nombre de ressources suffisant soit une domination d'une des deux espèces qui régulera le nombre de naissances de l'espèce dominée. Tous les graphes suivent les niveaux trophiques comme tels : producteur => herbivore => prédateur primaire => prédateur secondaire.

#### 2.1.3 Plaines

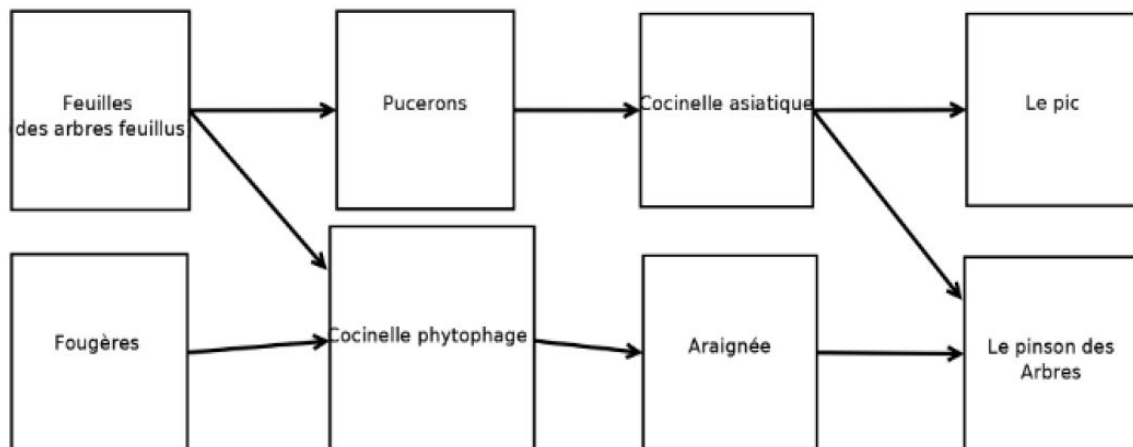


FIGURE 1 – Chaîne alimentaire des plaines

#### 2.1.4 Marécages

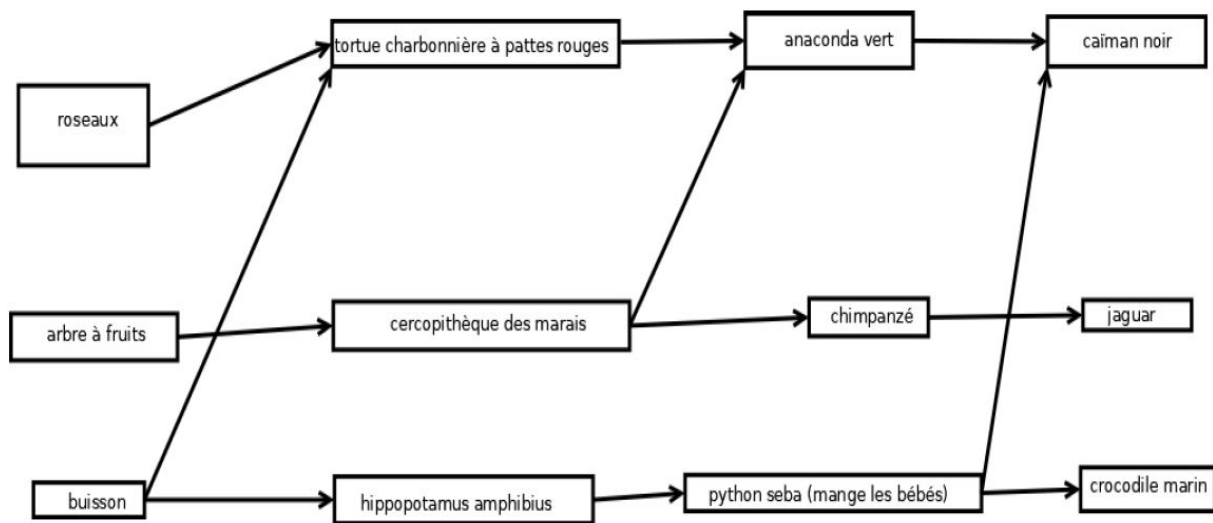


FIGURE 2 – Chaîne alimentaire des marécages

#### 2.1.5 Glacial

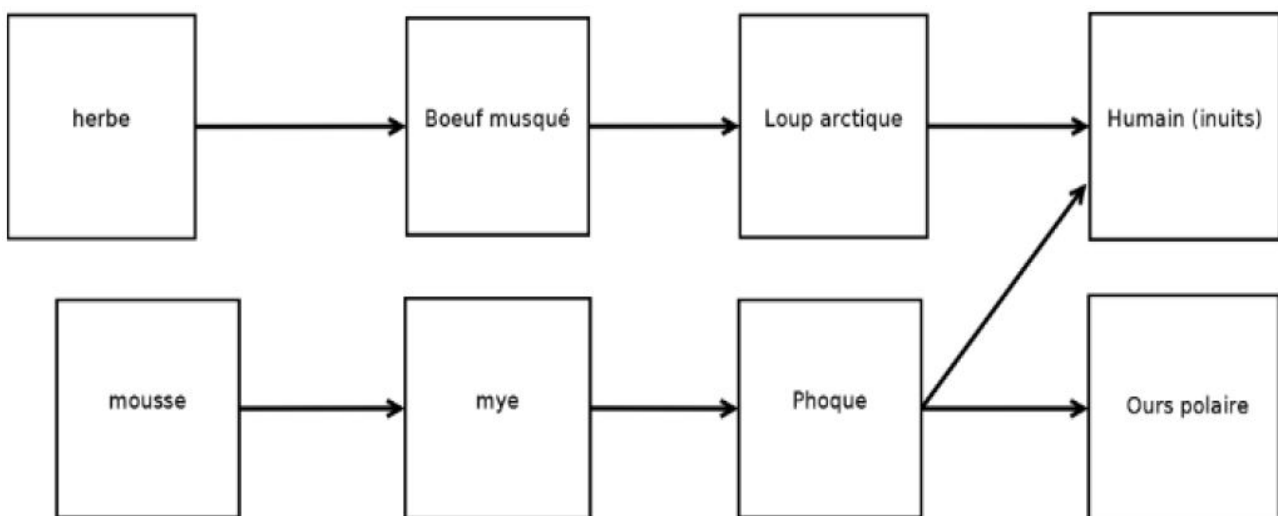


FIGURE 3 – Chaîne alimentaire de la banquise

### 2.1.6 Savane

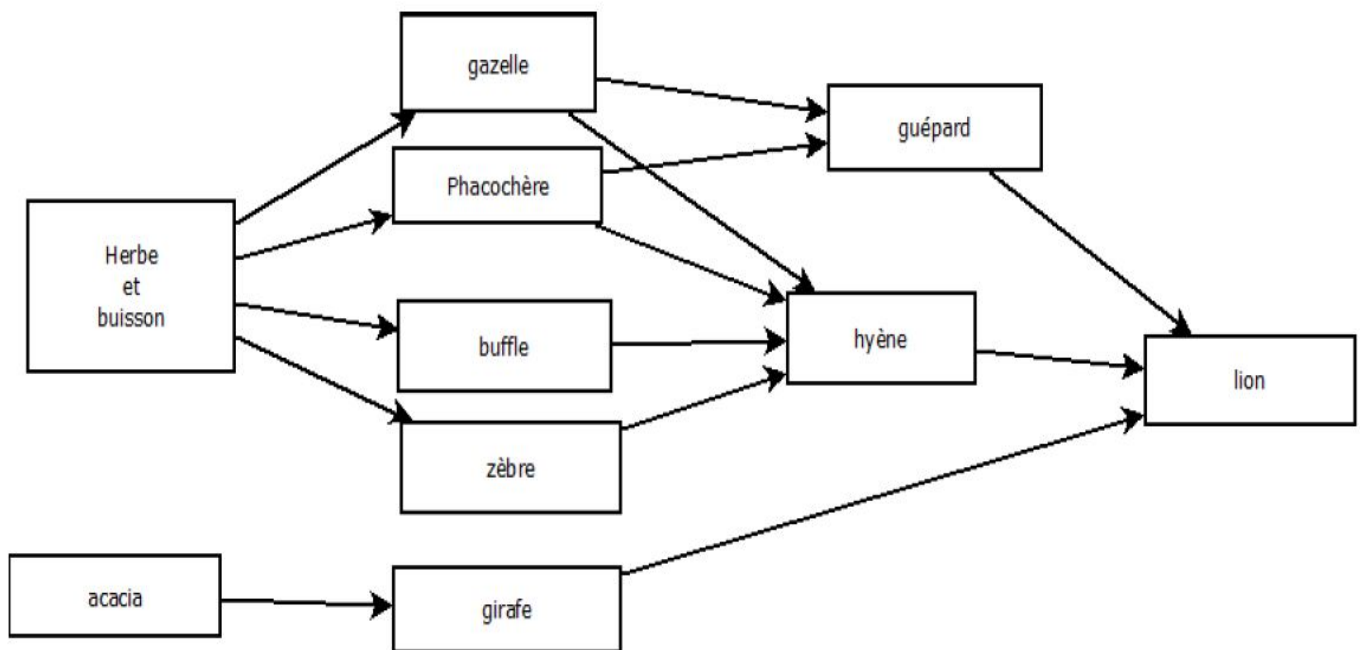


FIGURE 4 – Chaîne alimentaire de la savane

### 2.1.7 Lac

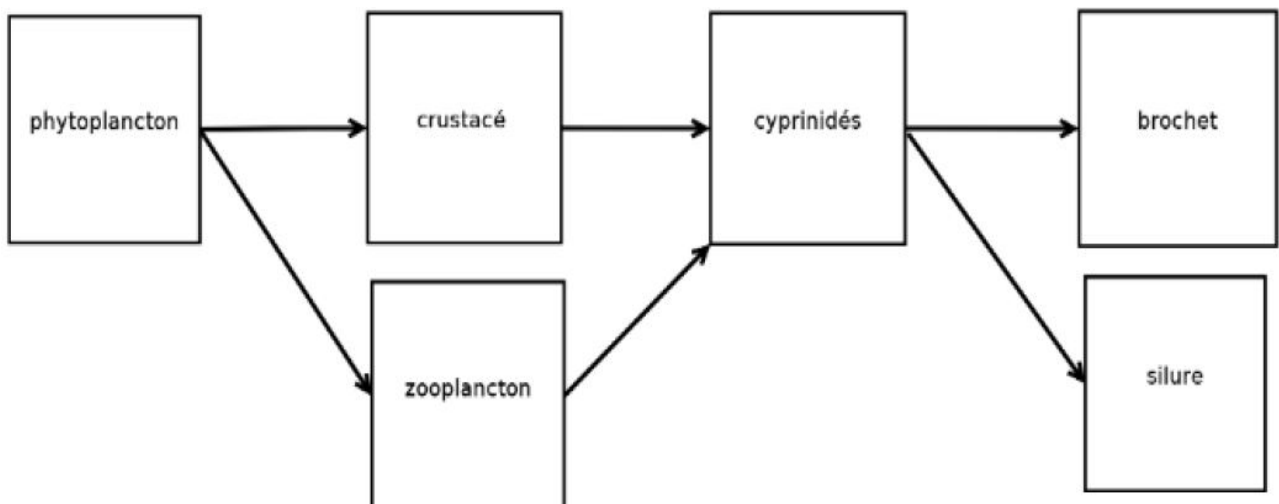


FIGURE 5 – Chaîne alimentaire du lac



Comme les espèces seront vouées à migrer, nous prévoyons de lier les chaînes alimentaires entre-elles après s'être concentrés sur un seul territoire.

#### **2.1.8 Décomposeurs**

Les décomposeurs se chargeront du « recyclage » des matières organiques. Voici comment ils se répartissent :

1. des bactéries qui s'occuperont à la fois des cadavres et des excréments dans le territoire glacial ;
2. des bactéries s'occuperont des cadavres et des mouches s'occuperont des excréments dans le territoires marécageux, dans les plaines et dans la savane ;
3. des bactéries s'occuperont des cadavres et des mollusques s'occuperont des excréments dans le territoire aquatique.

#### **2.1.9 Migrations**

Basiquement, les individus d'une même espèce restent groupés sur un lieu. Quand un individu a besoin de se nourrir il devra chercher autour de lui de quoi se nourrir. Si, dans un territoire, il ne reste pas assez de ressources ou si son espèce est en infériorité par rapport à une espèce concurrente, il y a des chance qu'un individu fasse le choix de migrer. Cette espèce dans son nouveau écosystème aura pour chaîne alimentaire : il mangera tous les espèces qui ont un niveau trophique inférieur à lui et son prédateur sera tous les espèces dont leur niveaux trophiques est supérieur à lui. En plus de cela, nous ajouterons des aléas tels que des incendies qui tueront des producteurs et une maladie qui baisse progressivement la santé d'un animal.

### **2.2 Fonctionnalités attendues**

Avant de lancer la simulation, l'utilisateur aura la capacité de paramétrer :

1. la vitesse de croissance d'un écosystème ;
2. la densité maximale des espèces .

Il lui sera aussi possible d'accélérer la vitesse de la simulation, mais aussi de consulter le nombre de « tours » ou encore de lancer un aléa.

En outre, il lui sera possible de visualiser diverses informations durant la simulation :

1. un affichage de l'effectif de chaque espèces dans l'écosystème ;
2. un moyen de connaître les chaînes alimentaires.

### 3 Conception et réalisation du projet

Après avoir réuni toutes les informations et tous les objectifs à réaliser, nous avons poursuivi sur la conception du logiciel.

#### 3.1 Architecture globale du logiciel

Dans la figure 6, on peut voir la représentation du logiciel sous une forme globale. Nous avons le noyau d'un côté et l'IHM graphique de l'autre. L'IHM graphique donnera vie au noyau, elle permettra de donner un visuel cohérent et informatif afin que la simulation soit la plus compréhensible. Quant au noyau, il contient toutes les données du logiciel tels que :

1. les données de toutes les espèces ;
2. les données liées aux coordonnées de chaque espèces dans un écosystème ;
3. les données de chaque écosystème.

De plus, il y a la partie traitement qui utilise les données.

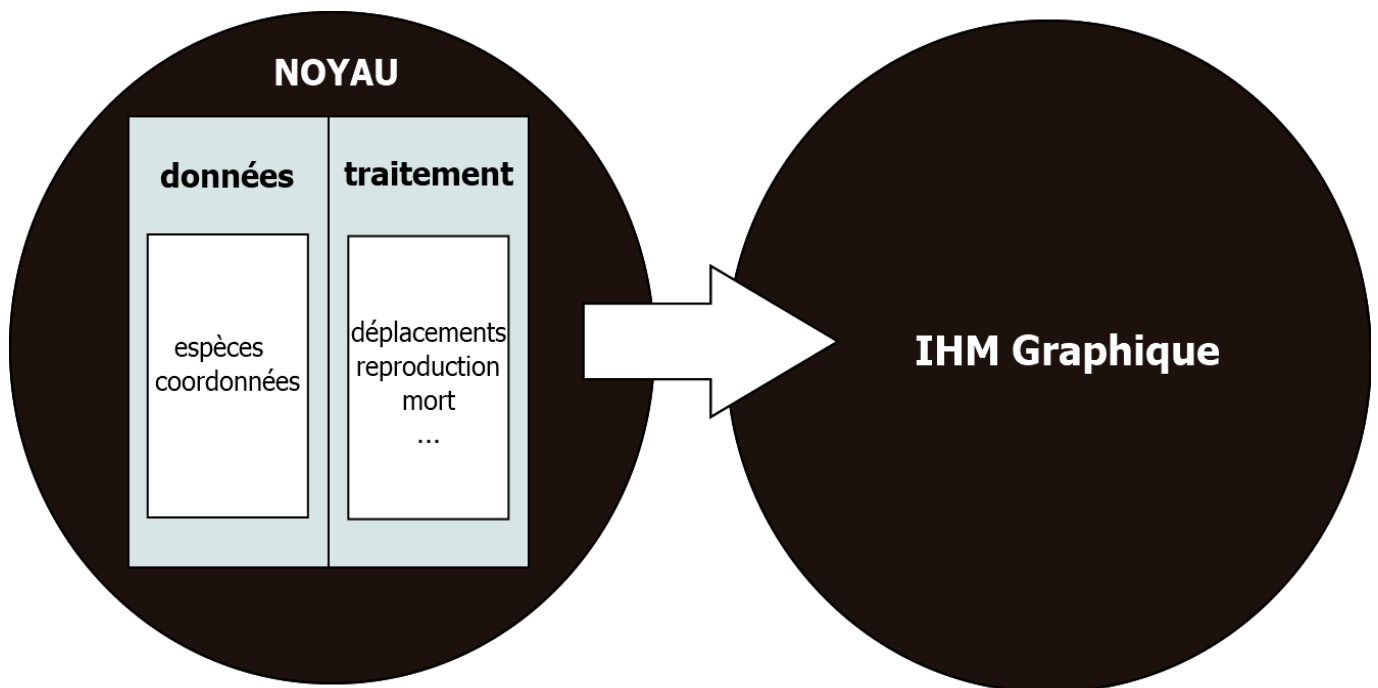


FIGURE 6 – Structure du logiciel

### 3.2 Conception des classes de données

La section des données contient l'arborescence logique des espèces dans des chaînes alimentaires.

La classe principale nommée Species est une interface qui implémente toutes les autres classes liées aux êtres vivants. Elle possède des getters et setters globaux :

1. la population dans un groupe ;
2. le nom de l'espèce ;
3. les coordonnées d'un groupe ;
4. si les êtres vivant sont en vie ou non ;
5. les points de vie d'un groupe ;
6. le nombre de jour pour qu'un être vivant se reproduise.

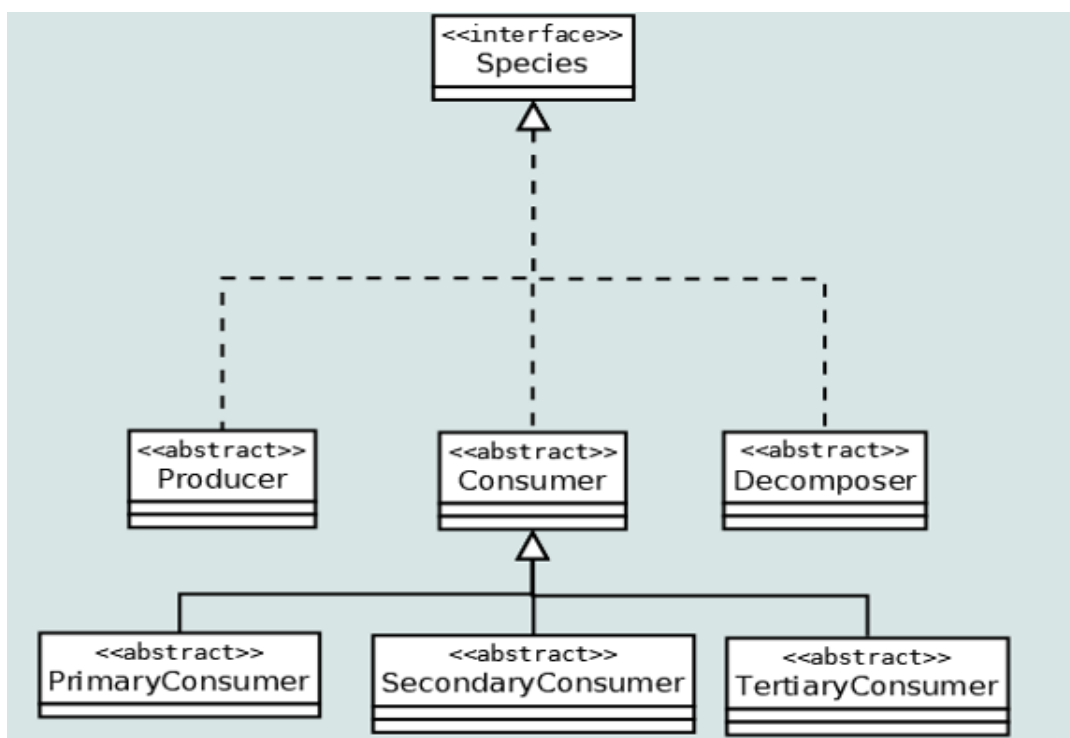


FIGURE 7 – UML du squelette des classes de données

Après avoir mis en place ce squelette, nous avons classé les espèces créés plus tôt dans leur niveau trophique.

Rangement par niveau trophique dans la savane	Hérite de la classe
Acacia	Producer
Grass	Producer
Bush	Producer
Gazelle	PrimaryConsumer
Warthog	PrimaryConsumer
Cheetah	SecondaryConsumer
Buffalo	PrimaryConsumer
Zebra	PrimaryConsumer
Hyena	SecondaryConsumer
Giraffe	PrimaryConsumer
Lion	TertiaryConsumer

TABLE 1 – Rangement des espèces de l'écosystème savane

De plus, chaque niveau trophique possède des attributs propres à eux-même. En somme, chaque classe d'espèce hérite de son niveau trophique et ne contient qu'un constructeur avec, en plus, une ArrayList de chaînes de caractères des espèces qu'il chasse et une autre des espèces qui le chassent. Tout cela nous a permis de rassembler les informations utiles pour la partie traitement du logiciel.

### 3.3 Conception des traitements (processus)

À l'issue de la création des classes de données, nous nous sommes attelés à conception des traitements du logiciel. Cette partie se divise en trois sous-parties, chacune regroupant des traitements sur des aspects communs.

#### 3.3.1 La gestion des espèces

Dans cette première branche nous nous sommes penchés sur la création d'êtres vivants. En utilisant un Singleton pattern. En effet, cela nous permet de ranger tous les êtres vivants d'un écosystème et des les classer en fonction de leur nom, plus précisément, il nous a fallu créer une HashMap ayant pour couple : <String , Species [ ]>.

En plus de la génération nous avons traité la reproduction. La reproduction est possible au bout d'un nombre de jour codé en dur, il se récupère dans les données de chaque espèce. Si le groupe d'individus se reproduit, alors le nombre d'individus augmente et leur taille est augmentée sur l'interface graphique durant la simulation. Cependant, il y a une limite d'effectif par groupe. Si un groupe dépasse dix individus, alors le surplus devient un nouveau groupe.

#### 3.3.2 La mort et la transformation minérale

Nous avons fait en sorte que si un groupe prédateur est à la même position qu'un groupe dont il peut se nourrir, le groupe mangé perd de l'effectif. Cela se produit seulement si les prédateurs ont besoin de se nourrir. De plus un groupe disparaît, s'il n'y a plus d'individus. L'endroit où des individus sont morts contiendra de la matière minérale qui sera retransmise aux producteurs.

#### 3.3.3 Le mouvement des espèces

Chaque jour, les espèces se déplacent. S'ils n'ont pas besoin de se nourrir alors ils effectuent un déplacement aléatoire dans une des quatre directions possibles (haut, bas, droite, gauche) et le tirage aléatoire est équitable.

À l'inverse, si les individus doivent se nourrir, alors elles analyseront toutes leurs proies sur l'écosystème pour connaître la plus proche, suite à cela, elle effectuera un déplacement en fonction de la direction ciblée.

### 3.4 Conception de l'IHM graphique

Notre logiciel se tient sur deux fenêtres. Il y a une fenêtre de sélection d'écosystèmes et une fenêtre mettant à l'image la simulation.

#### 3.4.1 Le menu de sélection d'écosystèmes

Cette fenêtre permet d'alterner entre les différents écosystèmes du logiciel, elle y contient quatre boutons.

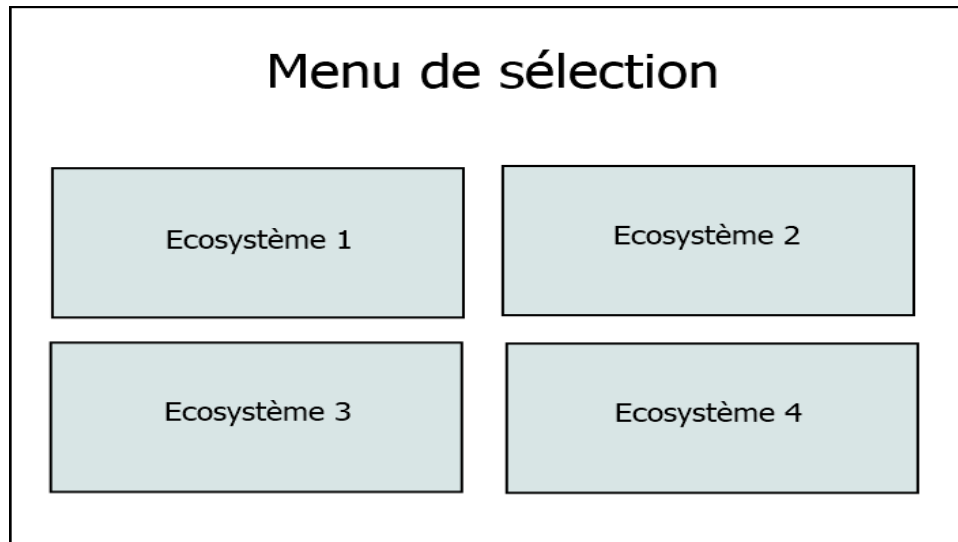


FIGURE 8 – Schéma de la fenêtre de sélection d'écosystèmes

Cette configuration modifiera l'affichage de la fenêtre de simulation.

#### 3.4.2 La fenêtre de simulation d'un écosystème

La fenêtre de simulation, fenêtre principale du logiciel se répartit en trois grandes parties : la zone de simulation où on y voit les êtres-vivants en action, la zone d'opérations afin d'agir sur la simulation et la zone d'informations permettant l'instrumentation du logiciel.

La zone de simulation affiche un écosystème avec des êtres-vivants en mouvement, à chaque itération on y voit les animaux se déplacer de cases en cases, mourir, se reproduire etc.

La zone d'opérations y contient des boutons d'action tels que les boutons d'aléas, événement permettant d'agir sur la simulation. Il y a aussi un bouton d'arrêt, un bouton de pause et de relancement.

La zone d'informations, permettant d'analyser les tenants et les aboutissants de la simulation contient deux parties majeures :

1. les informations sur les espèces en temps réel ;
2. les chaînes alimentaires.

Pour ce qui est des chaînes alimentaires, c'est un tableau de quatre colonnes :

1. nom de l'espèce ;
2. prédateur(s) de l'espèce ;
3. proie(s) de l'espèces ;
4. niveau trophique.

C'est un tableau qui ne changera pas, les informations sont statiques et elles ne sont là que pour appuyer la propension des prédateurs à rejoindre une espèce pour se nourrir.

Un autre tableau présente :

1. le nom de l'espèce ;
2. le nombre de groupes de cette espèce dans l'écosystème ;
3. la population totale ;
4. les groupes affamés ;
5. le niveau trophique de l'espèce.

Enfin, il y a un graphique représentant le nombre d'animaux en vie de chaque niveau trophique.

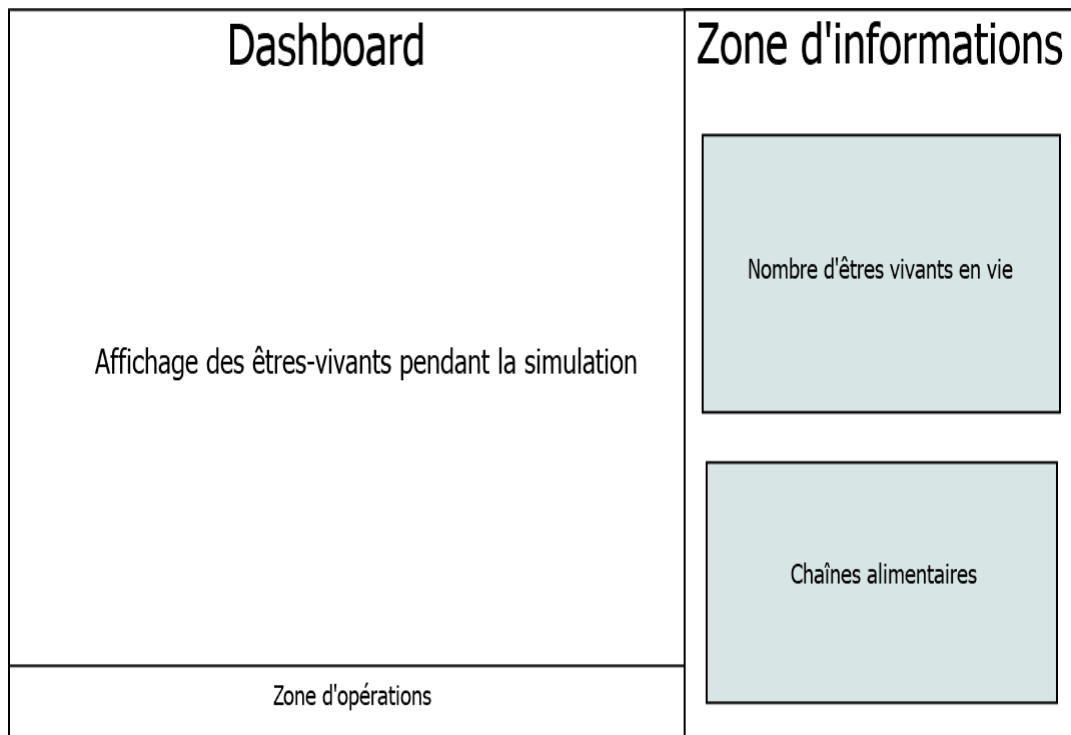


FIGURE 9 – Schéma de la fenêtre de simulation

## 4 Manuel utilisateur

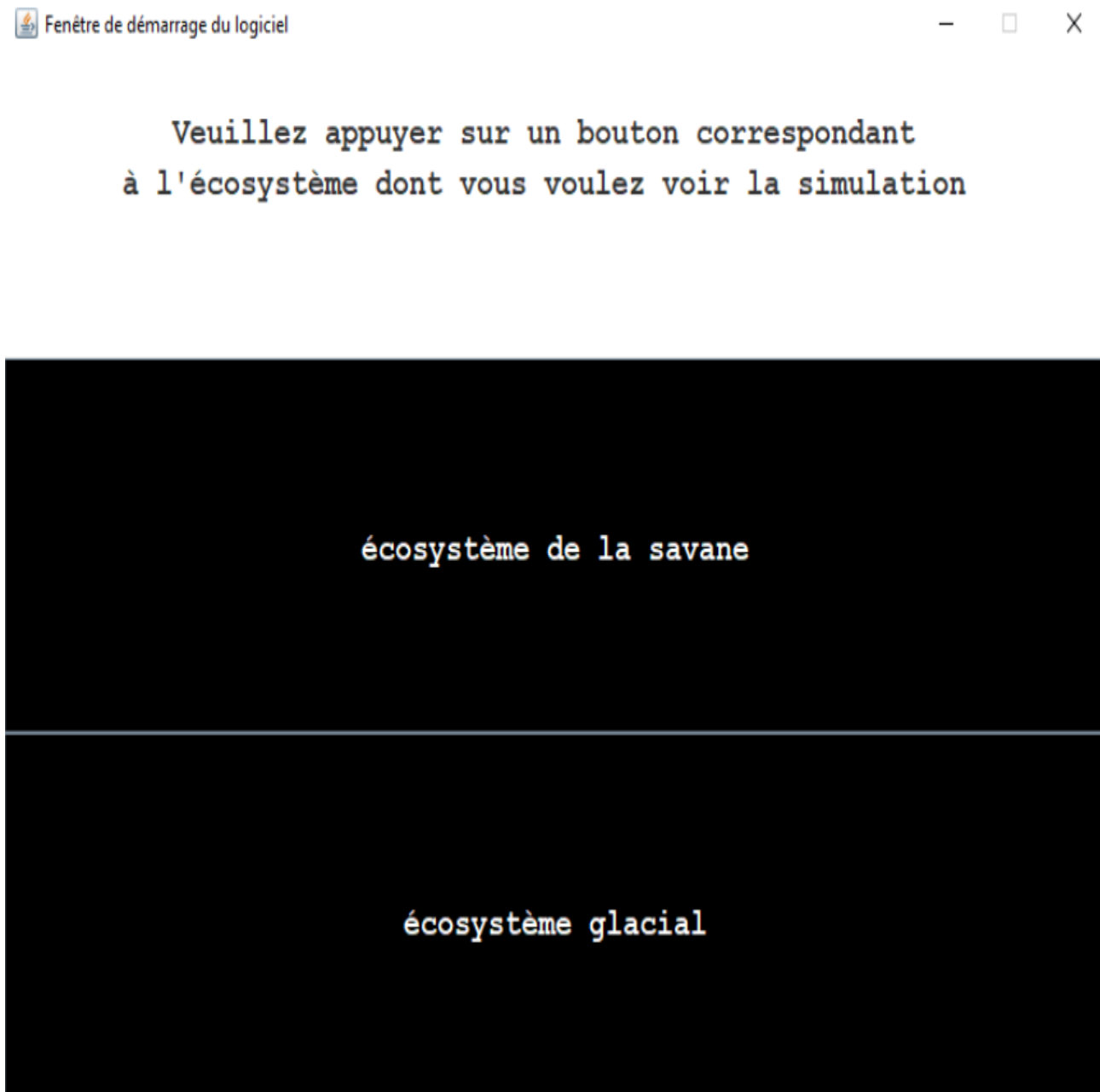


FIGURE 10 – Fenêtre de démarrage

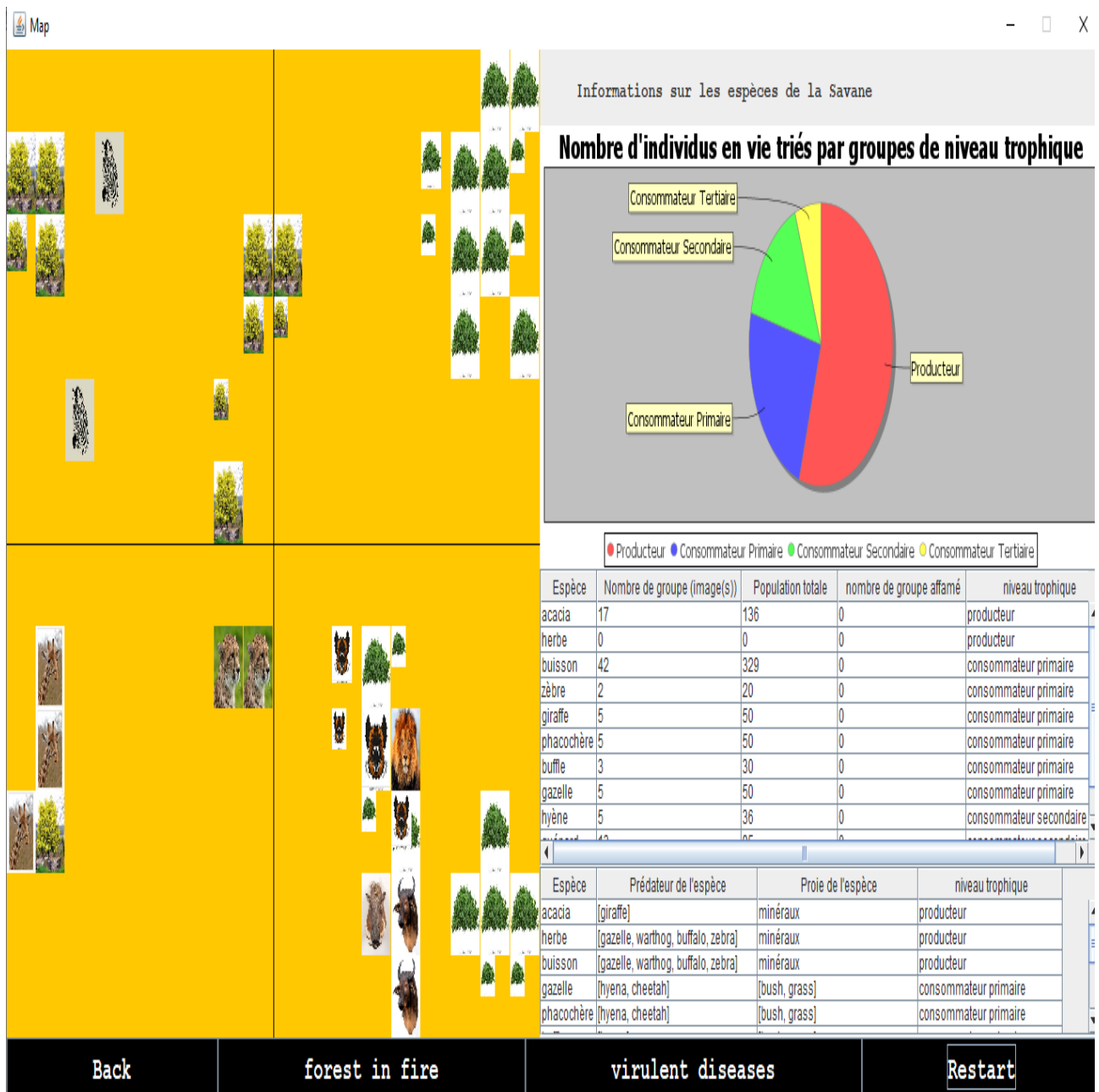


FIGURE 11 – fenêtre de simulation



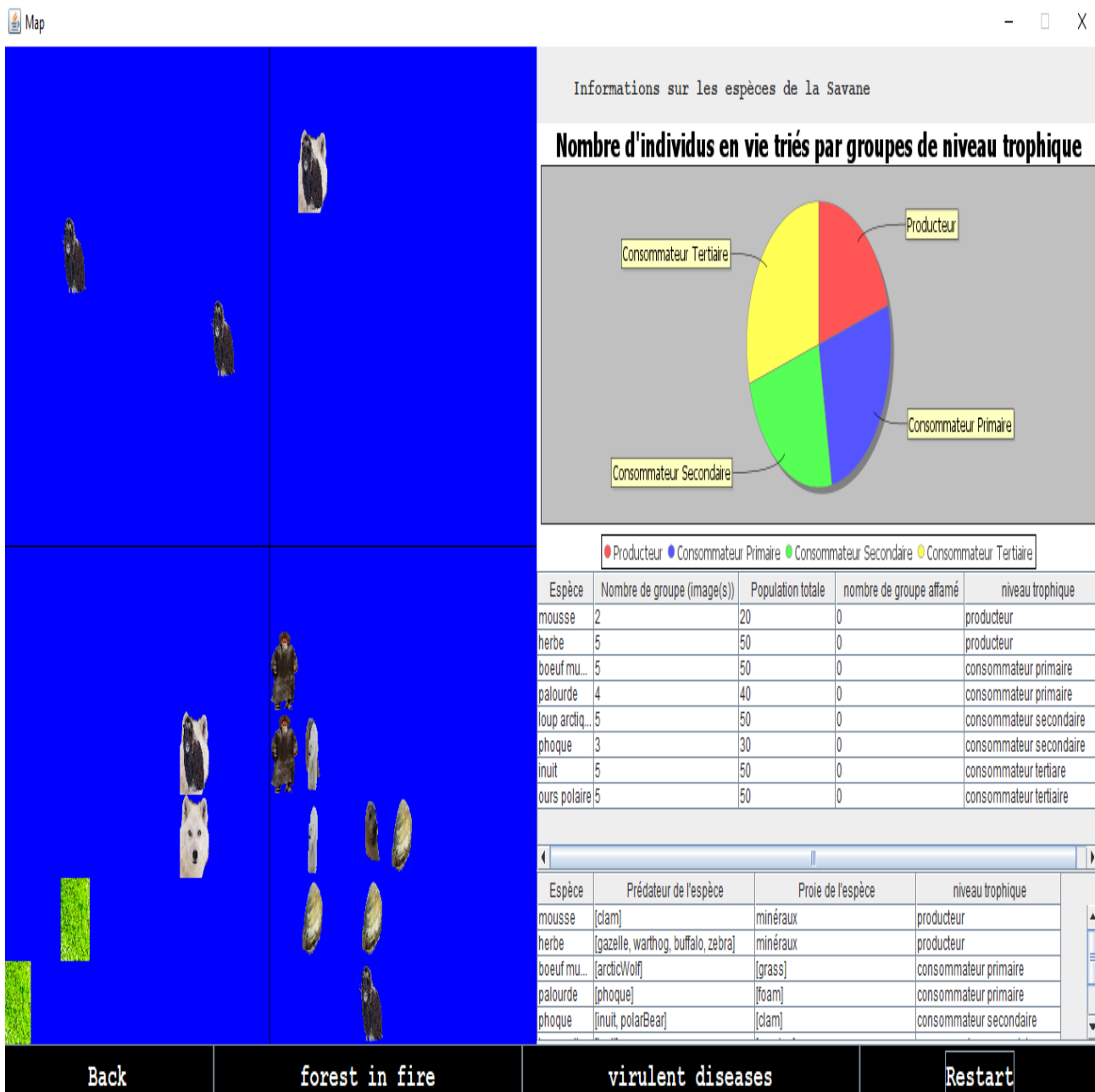


FIGURE 12 – Simulation de l'écosystème

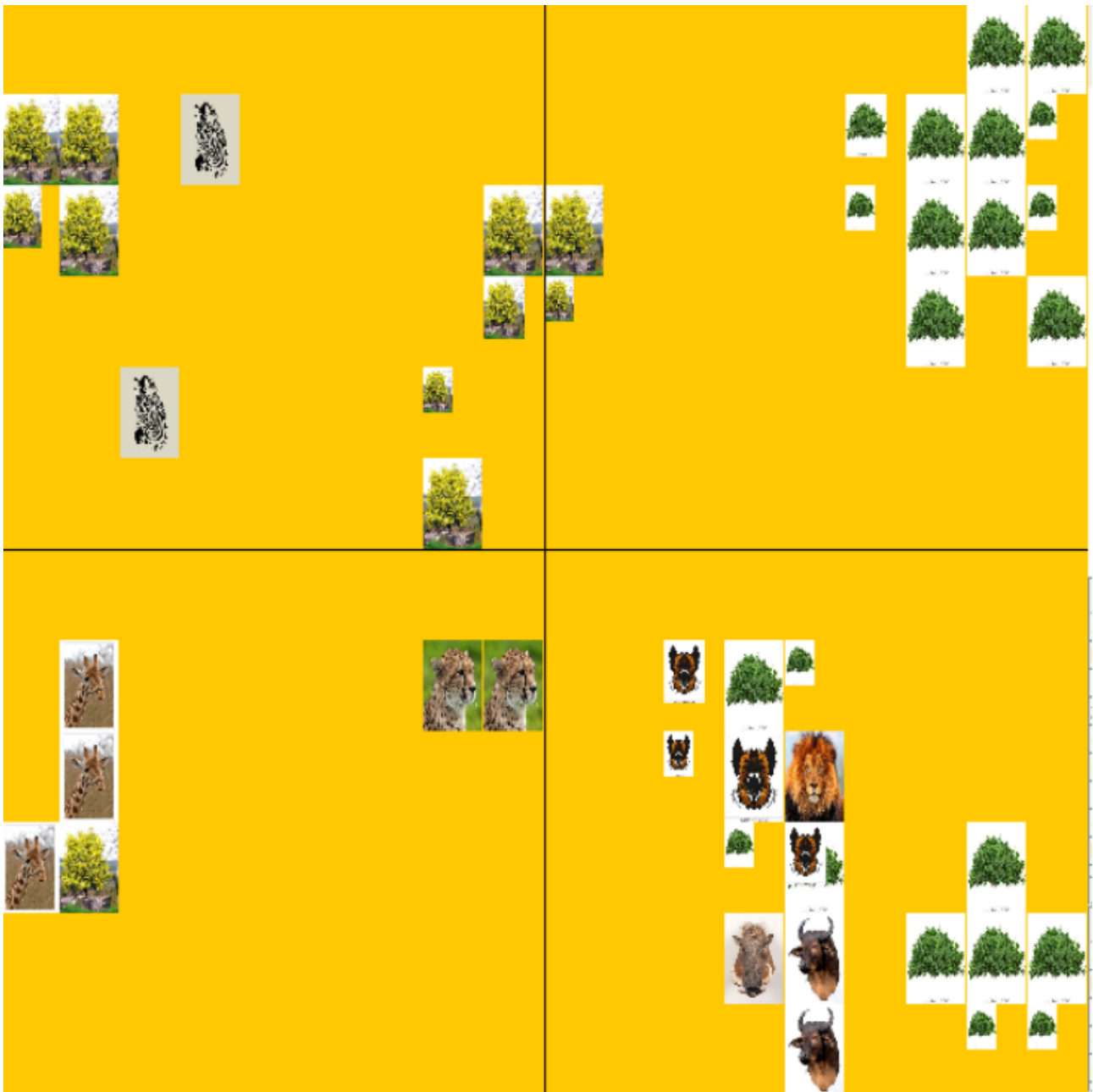


FIGURE 13 – Zone de simulation

## Nombre d'individus en vie triés par groupes de niveau trophique

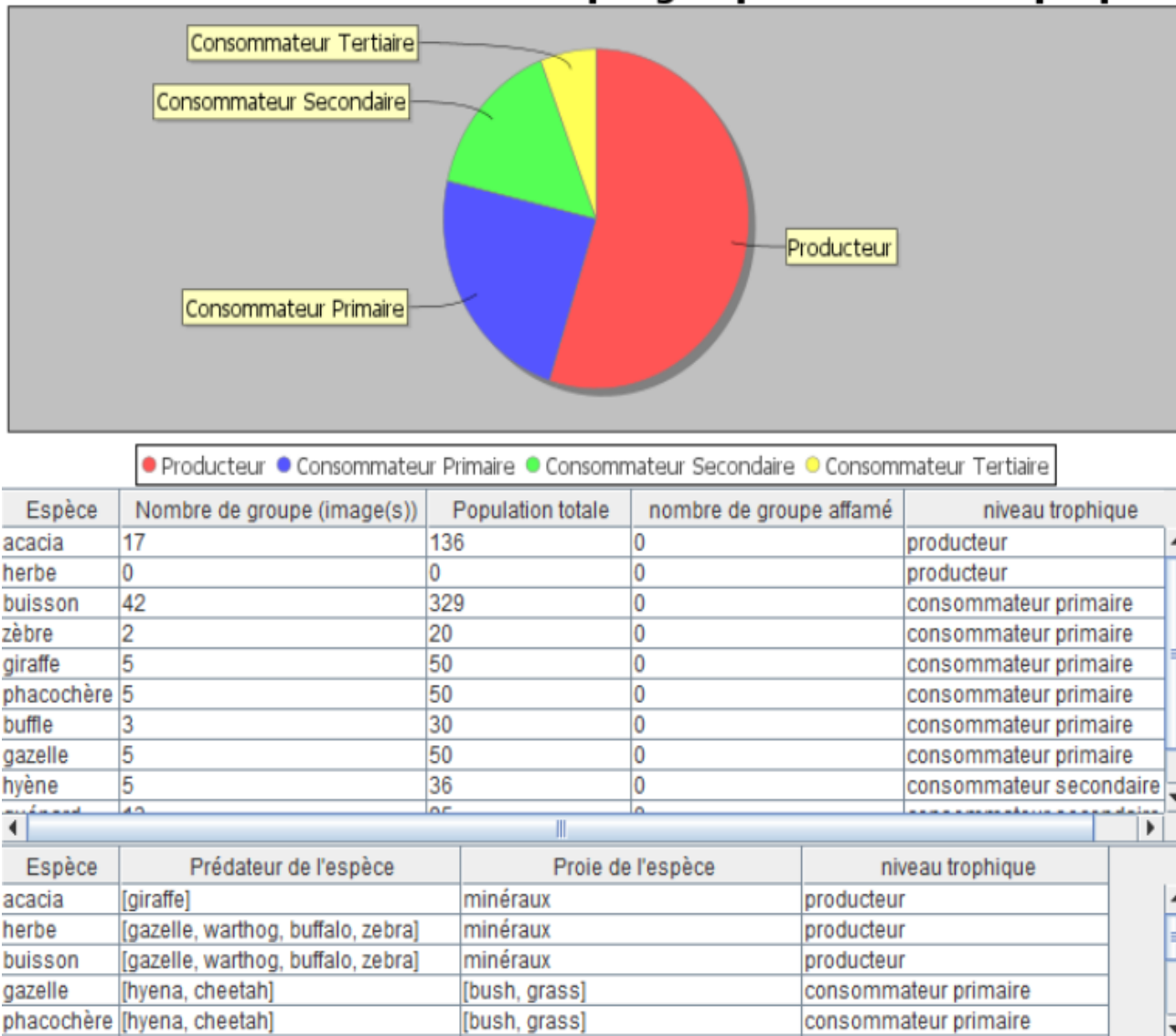


FIGURE 14 – Partie informations

Lorsque nous voyons la fenêtre de démarrage du logiciel, on appuie sur un des deux boutons qui nous emmène sur l'écosystème souhaiter, dans le panel information pour afficher le JFreechart on doit appuyer sur stop. on peut revenir en arrière en appuyant sur back, ce qui nous permet de revenir en arrière et de changer d'écosystème. le bouton "forest in fire" tue le moitié des producteurs dans un écosystème et le bouton virulent disease tue la moitié des consommateurs. L'écosystème Glacial ne marche qu'à moitié.

## 5 Déroutement du projet

Dans cette section, nous décrivons comment le projet a été réalisé en équipe : la répartition des tâches, la synchronisation du travail en membres de l'équipe, etc.

### 5.1 Réalisation du projet par étapes

La réalisation du projet s'est répartie de nombreuses étapes distinctes.

1. Lecture minutieuse du sujet du projet, repérage des éléments devant être présent dans notre programme, repérage des éléments qui sont optionnel dans le programme, écriture du cahier des charges pour savoir ce qu'il faut faire dans l'ordre, qui pourrait faire quoi. Le cahier des charges est sujet à des modifications au cours du temps.(N.B. Le diagramme de GANTT qui est proposé ci-dessous est celui qui possède la mise à jour la plus récente)
2. En premier, des recherches on été faites sur les définitions scientifiques de chaque mots comme par exemple écosystème, chaîne alimentaire, dans le but de bien comprendre notre sujet et de ne pas faire de hors-sujet. Après ces enseignements rapide sur la science de la vie, nous avons formé des chaînes Alimentaires simples pour notre simulation. Ces quelques chaînes simplifiées nous ont permis de créer une base de donnée.
3. Le traitement a permis de relier les données entre elles pour procéder à la traduction des données scientifiques en algorithme.
4. Pour L'IHM Graphique, on a commencé par faire un croquis/schéma de notre fenêtre qui servi de base pour le codage et la conception.
5. Nous avons pris notre temps pour réparer les problèmes qui ont été remarqués.

### 5.2 Répartition des tâches entre membres de l'équipe

La répartition des tâches s'est faite au début du projet et n'a cessé d'évoluer tout au cours de sa réalisation.

La compréhension du sujet, la création du cahier des charges et des classes de donnée ont été faites ensemble.

Pour le cas de la partie traitement, nous nous étions répartis les tâches en fonction des types de traitements à produire.



FIGURE 15 – Diagramme de répartition des tâches

La répartition nous a permis d'assimiler les délais afin de finir le projet à temps.

## 6 Conclusion et perspectives

Dans cette section, nous résumons la réalisation du projet et nous présentons également les extensions et améliorations possibles du projet.

### 6.1 Résumé du travail réalisé

C'est la première fois que nous réalisons un projet complexe. De l'avis général, nous avons consolidé nos connaissances générales et appris à faire des applications plus attrayantes et plus orientées pour le monde du travail. Nous sommes globalement satisfaits de ce que nous avons réalisé malgré les quelques défauts et le confinements qui est arrivé au milieu du semestre.

Nous avons réalisé un programme qui simule un écosystème comme il nous a été demandé dans le sujet. Ce projet a été créé dans le but de faire une simulation qui se rapproche le plus d'une simulation scientifique. Ce programme possède un algorithme complexe qui regroupe la plupart des instruction qui nous a été donné durant les cours de Génie logiciel et projet(GLP).

### 6.2 Améliorations possibles du projet

Ajout d'une fonctionnalité permettant à un utilisateur de personnaliser le début de la simulation en fonction des envie de l'utilisateur :

- choix des espèces qu'il pourrait mettre dans l'écosystème,
- choix du nombre de la population pour chaque espèce au début de la simulation,
- choix de la chaine alimentaire.

Ajout de plusieurs autres choix différents pour les désastres naturels.

Ajout de plusieurs autres écosystèmes différents

Améliorations de l'affichage de la carte montrant les espèces :

- changement de l'image de l'espèce par une autre qui est mieux adapté,
- fluidité du graphique quand on passe d'un tour à un autre.

## Références

[Fou] CK-12 Foundation. Food chains and food webs - advanced.

[Liu] Tianxiao Liu. Informations et documents pour le module génie logiciel.