



Simulation Du Comportement Des Fourmis

Mme Murisasco – Mr Francicsi

LEGRAND Vincent
MERLIN Lucas
PALMA François

Sommaire

I - Introduction

II – La conception UML

1/ L'axe fonctionnel

2/ L'axe statique

3/ L'axe dynamique

III – Manuel Utilisateur

1/ Les Menus

2/ Les Boutons

3/ La Simulation

IV – Conclusion

V - Annexe

Introduction

Lors du premier TP présentiel avec Mr FRANCICSI, nous, Vincent LEGRAND, Lucas MERLIN et François PALMA avons décidé de nous mettre en groupe afin de réaliser ce projet. En effet, lors de précédents TP, nous avons déjà communiqué et réfléchi ensemble, ce qui nous a permis de savoir que nous étions dans la capacité de travailler à trois.

Le sujet de ce projet était de réaliser la conception UML puis d'implémenter une application qui consiste à simuler le comportement des fourmis en fonction de certains facteurs.

Nous avons commencer à réfléchir sur les différents facteurs que nous voulions retenir et avons fixé les premières exigences.

Ces premières réflexions nous ont permis de réaliser que nous avions chacun des visions bien différentes sur certains sujets. Nous avons donc, dès le début, pris comme méthode de travail, de débattre sur un point afin de pouvoir garder les meilleurs idées de chacun et ensuite de se répartir le travail afin de ne pas partir sur une idée, chacun de son côté, et se retrouver avec plusieurs parties incohérentes.

Nous avons donc tout les trois travaillé sur tout les aspects du projet.

Conception UML

L'Axe Fonctionnel

Les Exigences (*a retrouver en annexe)

Diagramme de Contexte Statique :

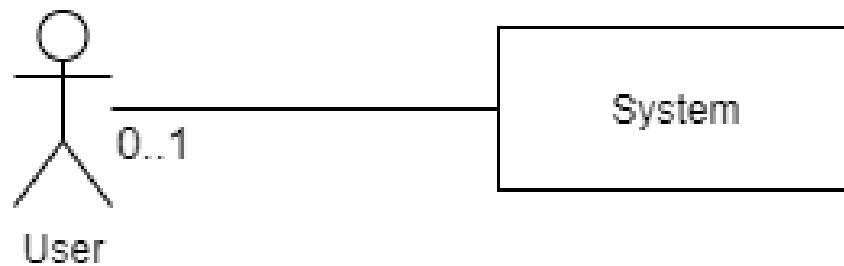
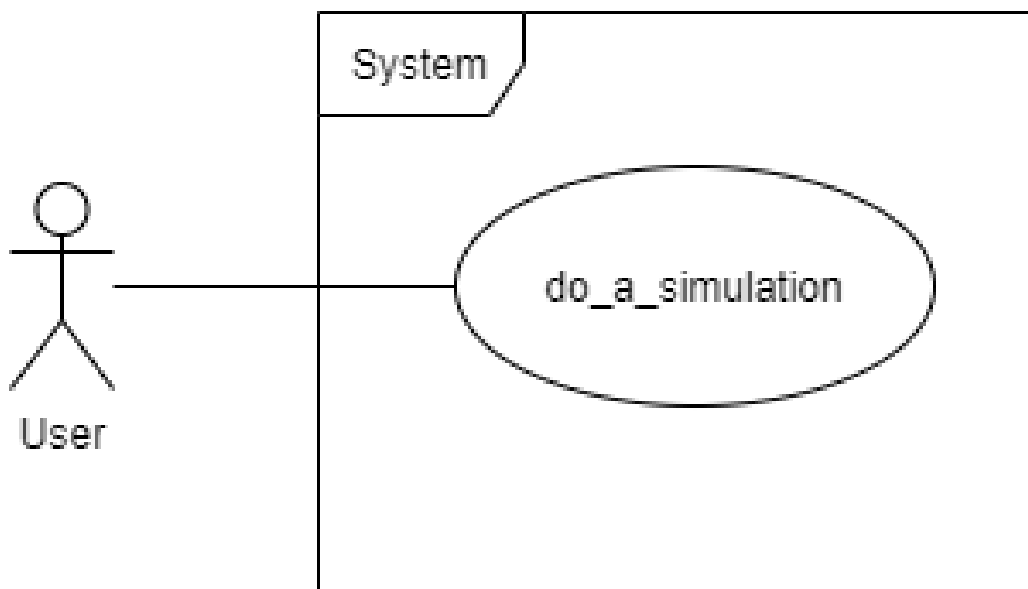


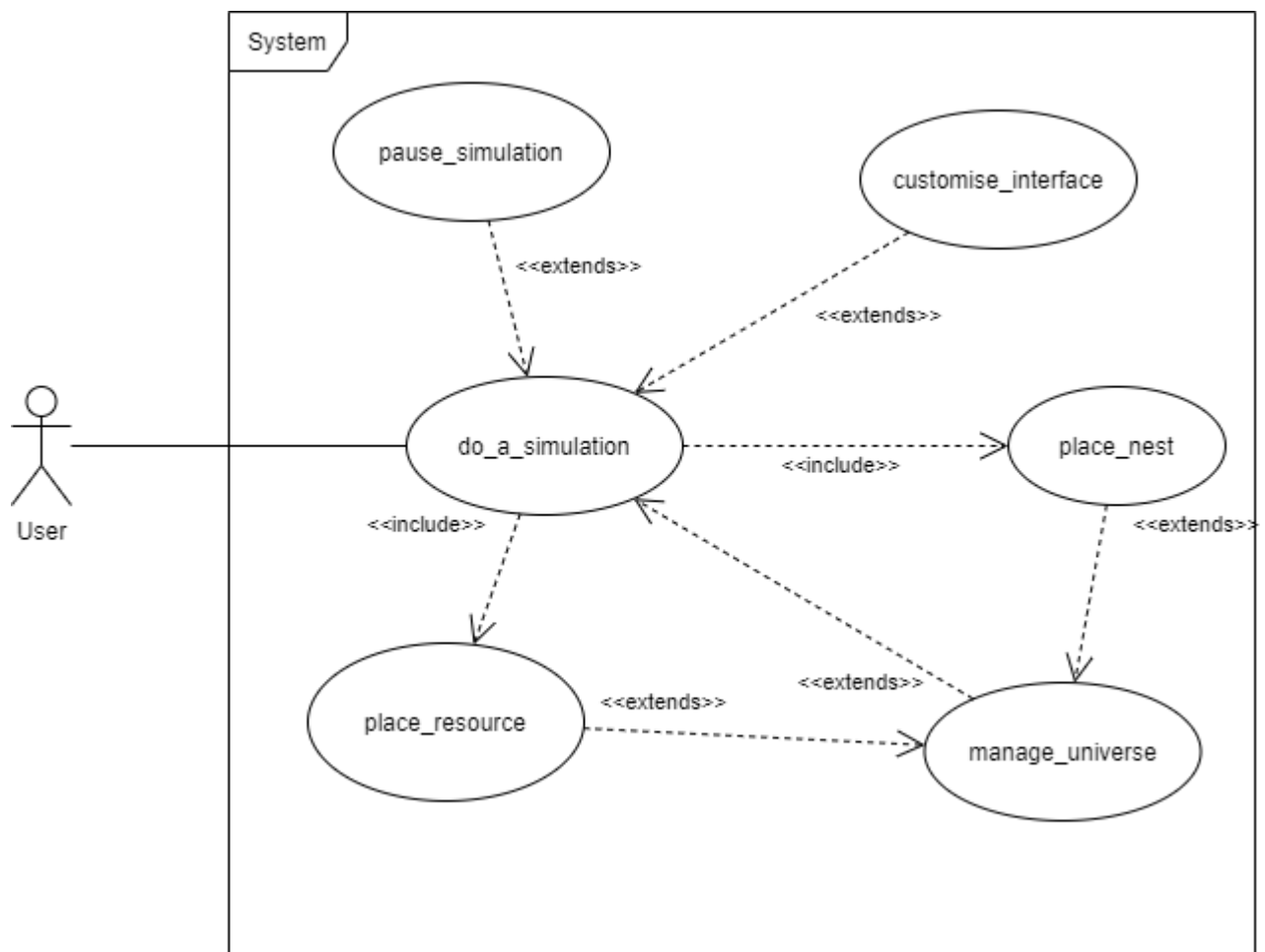
Diagramme de Cas d'Utilisation : (niveau 1)



Conception UML

L'Axe Fonctionnel

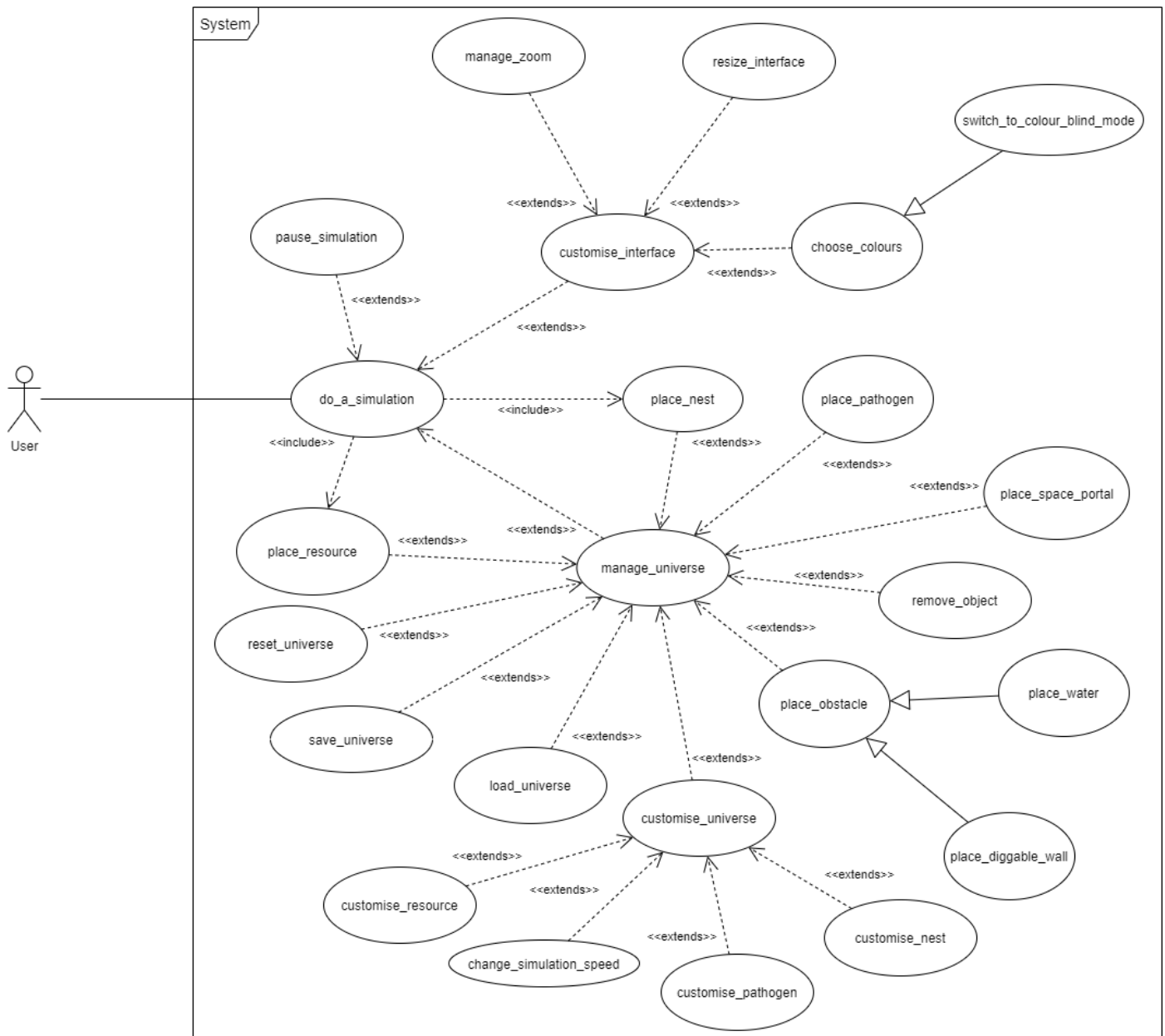
Diagramme de Cas d'Utilisation : (niveau 2)



Conception UML

L'Axe Fonctionnel

Diagramme de Cas d'Utilisation : (niveau 3)



Conception UML

L'Axe Fonctionnel

Scénario de Cas d'Utilisation :

Scenario Nominal : do_a_simulation

Résumé : ce cas permet à l'utilisateur de faire une simulation.

Acteurs : utilisateur, système

pré-condition : aucune

post-condition : aucune

1. L'utilisateur indique qu'il souhaite lancer une simulation
2. L'utilisateur pose autant de nid que souhaité
3. Le système affiche autant de nid que l'utilisateur a posé
4. l'utilisateur pose autant de ressources que souhaité
5. le système affiche autant de ressources que l'utilisateur a posé
6. L'utilisateur lance la simulation
7. Le système effectue la simulation

Alternatives :

A1. place_pathogen

Résumé : ce cas permet à l'utilisateur de poser des pathogènes afin de simuler une transmission de maladie. Commence après le point 5.

- 6a. L'utilisateur pose autant de pathogènes que souhaité
- 7a. le système affiche autant de pathogènes que l'utilisateur a posé
- 8a. Reprise au point 6

A2. customise_pathogen

Résumé : ce cas permet à l'utilisateur de paramétrer les pathogènes afin de moduler le taux d'infection, la période d'incubation et la létalité. Commence après le point 7a.

- 8b. L'utilisateur paramètre les pathogènes comme il le souhaite
- 9b. Le système effectue les changements sur les pathogènes
- 10b. Reprise au point 6

A3. customise_resource

Résumé : ce cas permet à l'utilisateur de paramétrer les ressources afin de moduler la quantité de nourriture présente dans l'univers simulé. Commence après le point 5.

- 6c. L'utilisateur paramètre les ressources comme il le souhaite
- 7c. Le système effectue les changements sur les ressources
- 8c. Reprise au point 6

A4. customise_nest

Résumé : ce cas permet à l'utilisateur de paramétrer les nids afin de moduler le nombre de fourmis initial, l'intervalle entre création de fourmis supplémentaires ou encore la quantité de nourriture consommé par la création d'une fourmis. Commence après le point 5.

- 6d. L'utilisateur paramètre les nids comme il le souhaite
- 7d. Le système effectue les changements sur les nids
- 8d. Reprise au point 6

Conception UML

L'Axe Fonctionnel

A5. change_simulation_speed

Résumé : ce cas permet à l'utilisateur de paramétrer la vitesse de simulation de l'univers. Commence après le point 5.

6e. L'utilisateur paramètre la vitesse de la simulation comme il le souhaite

7e. Le système effectue le changement de vitesse de la simulation

8e. Reprise au point 6

A6. remove_object

Résumé : ce cas permet à l'utilisateur d'effacer un objet posé dans l'univers, que ce soit un obstacle un nid ou une ressource. Commence après le point 5.

6f. L'utilisateur efface autant de composantes que souhaité

7f. Le système supprime autant de composantes que l'utilisateur a effacé

8f. Reprise au point 6

A7. place_obstacle

Résumé : ce cas permet à l'utilisateur de poser un obstacle du type souhaité, qu'il s'agisse d'un mur, d'un mur creusable ou d'un cours d'eau. Commence après le point 5.

6g. L'utilisateur choisit le type d'obstacle voulu

7g. L'utilisateur pose autant de composantes que souhaité

8g. Le système pose autant d'obstacles que souhaité

9g. Reprise au point 6

A8. load_universe

Résumé : ce cas permet à l'utilisateur de reprendre un univers enregistré auparavant. Commence après le point 1.

2h. L'utilisateur choisit un univers à charger

3h. Le système affiche l'univers choisi

4h. Reprise au point 2

A9 : reset_universe

résumé : ce cas permet à l'utilisateur de réinitialiser l'univers de la simulation. Commence après le point 5.

6i. L'utilisateur indique qu'il souhaite réinitialiser l'univers

7i. Le système réinitialise l'univers

8i. Reprise au point 6.

A10 : save_universe

résumé : ce cas permet à l'utilisateur de sauvegarder l'univers d'une simulation en cours afin de la reprendre plus tard ou d'effectuer des modifications temporaires avec possibilité de revenir sur un état précédent plus tard. Commence après le point 6.

7j. L'utilisateur indique qu'il souhaite sauvegarder l'univers

8j. Le système demande à l'utilisateur de préciser le nom du fichier et son lieu de sauvegarde

9j. L'utilisateur indique un nom de fichier et un lieu d'enregistrement pour le fichier à sauvegarder

10j. Le système procède à la sauvegarde selon les informations précisées par l'utilisateur

11j. Reprise au point 7

A11 : manage_zoom

Résumé : ce cas permet à l'utilisateur de zoomer/dézoomer sur l'interface de la simulation. Commence après le point 1.

2k. L'utilisateur indique le niveau de zoom souhaité

3k. Le système zoom/dézoom sur la simulation

4k. Reprise au point 2

Conception UML

L'Axe Fonctionnel

A12 : resize_interface

Résumé : ce cas permet à l'utilisateur de redimensionner l'interface. Commence après le point 1.

2l. L'utilisateur indique la taille de fenêtre souhaité

3l. Le système redimensionne la fenêtre

4l. Reprise au point 2

A13: choose_colours

Résumé : ce cas permet à l'utilisateur de paramétrer les couleurs des éléments de l'univers, tels que les nids ou les ressources. Commence après le point 5.

6m. L'utilisateur indique qu'il souhaite modifier une couleur d'une composante

7m. Le système change la couleur de la composante.

8m. Reprise au point 6

A14: switch_to_colour_blind_mode

Résumé : ce cas permet à l'utilisateur de passer en mode daltonien. Commence après le point 1.

2n. L'utilisateur indique qu'il souhaite activer le mode daltonien

3n. Le système active le mode daltonien

4n. Reprise au point 2

A15: pause_simulation

Résumé : ce cas permet à l'utilisateur de rajouter des éléments après coup. Commence après le point 7.

8o. L'utilisateur indique au système qu'il souhaite interrompre la simulation.

9o. Le système interrompt la simulation

10o. Reprise au point 2.

A16: place_space_portal

Résumé: ce cas permet à l'utilisateur de placer un portail spatial et de relier spatialement deux points de l'univers simulé. Commence après le point 5.

6p. L'utilisateur pose autant de portails que souhaité par paires de 2

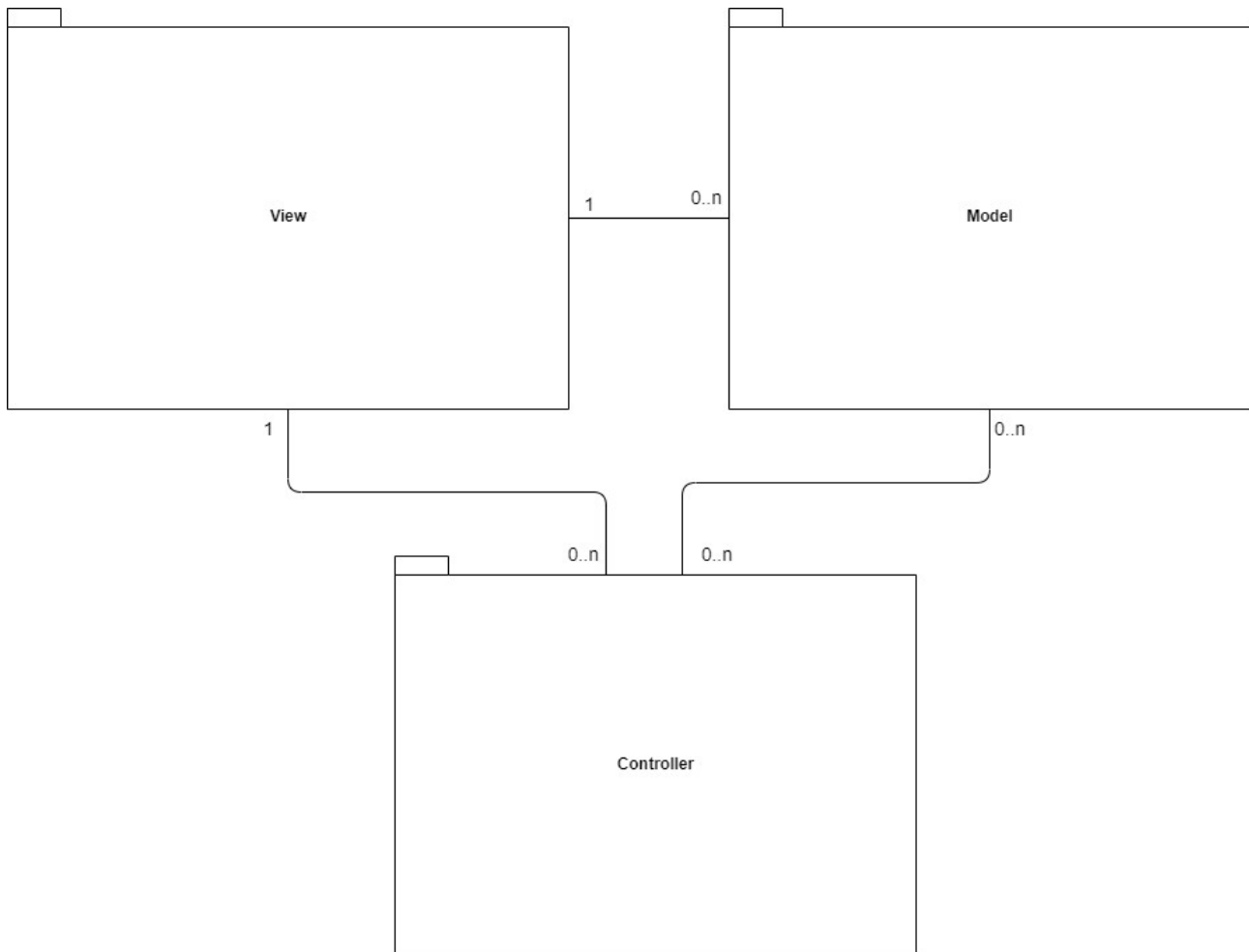
7p. Le système affiche les portails en conséquence

8p. Reprise au point 6

Conception UML

L'Axe Statique

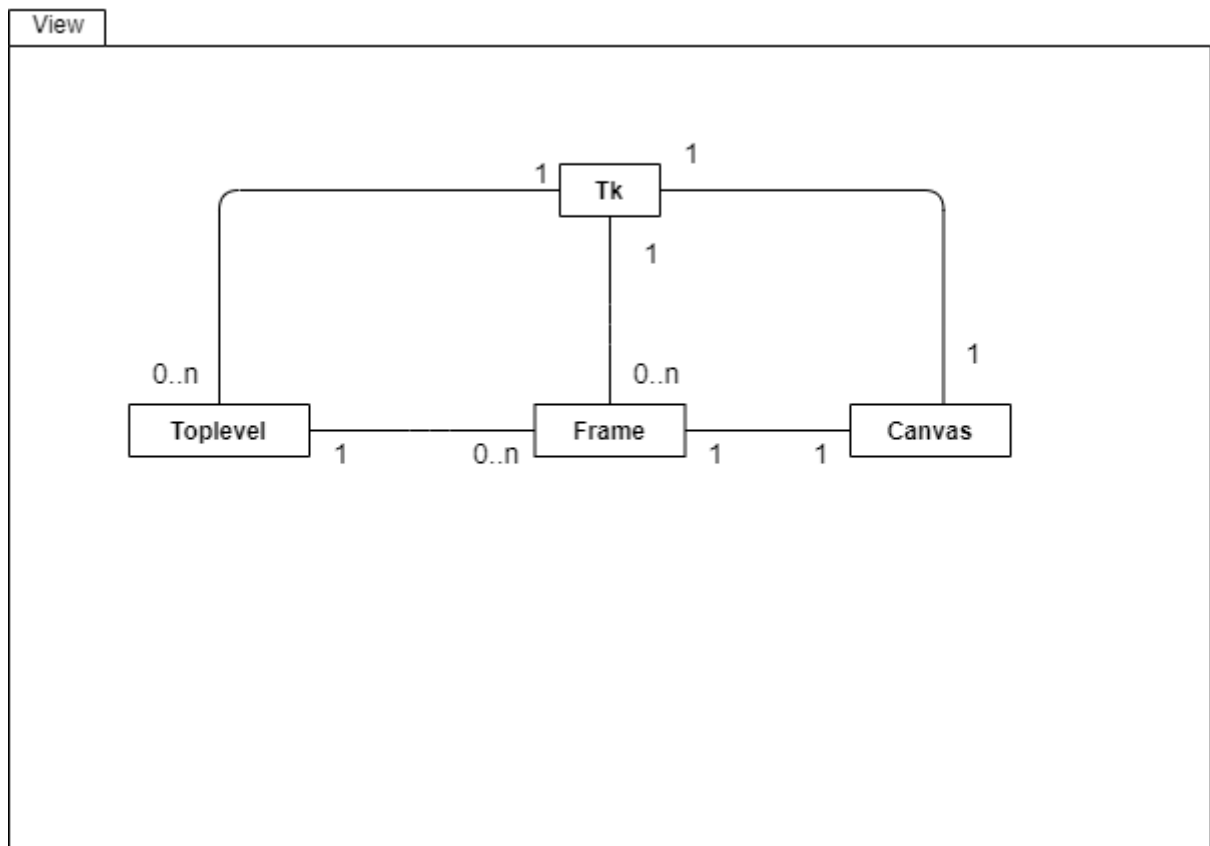
Diagramme de Paquets :



Conception UML

L'Axe Statique

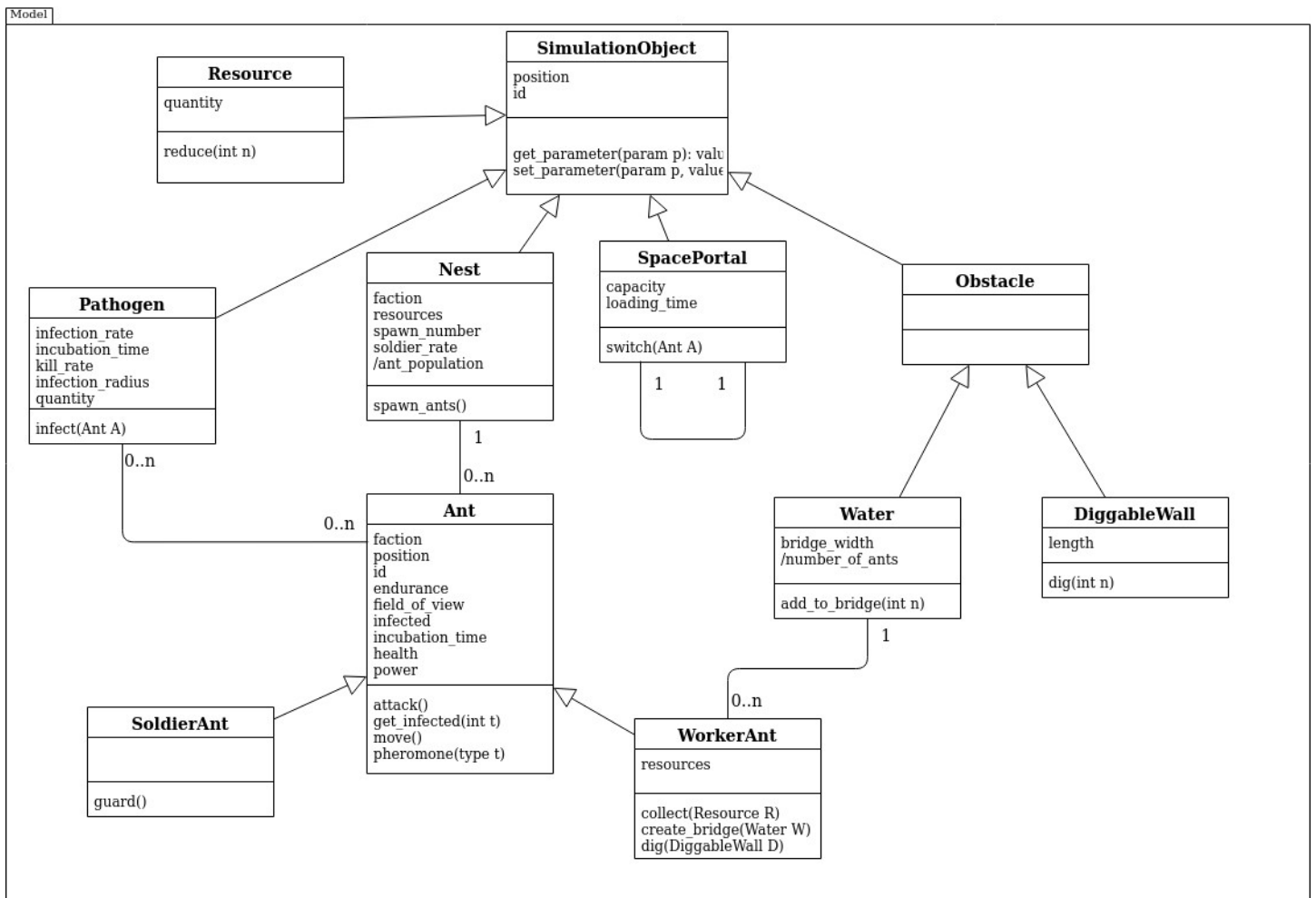
Diagramme du Paquet View :



Conception UML

L'Axe Statique

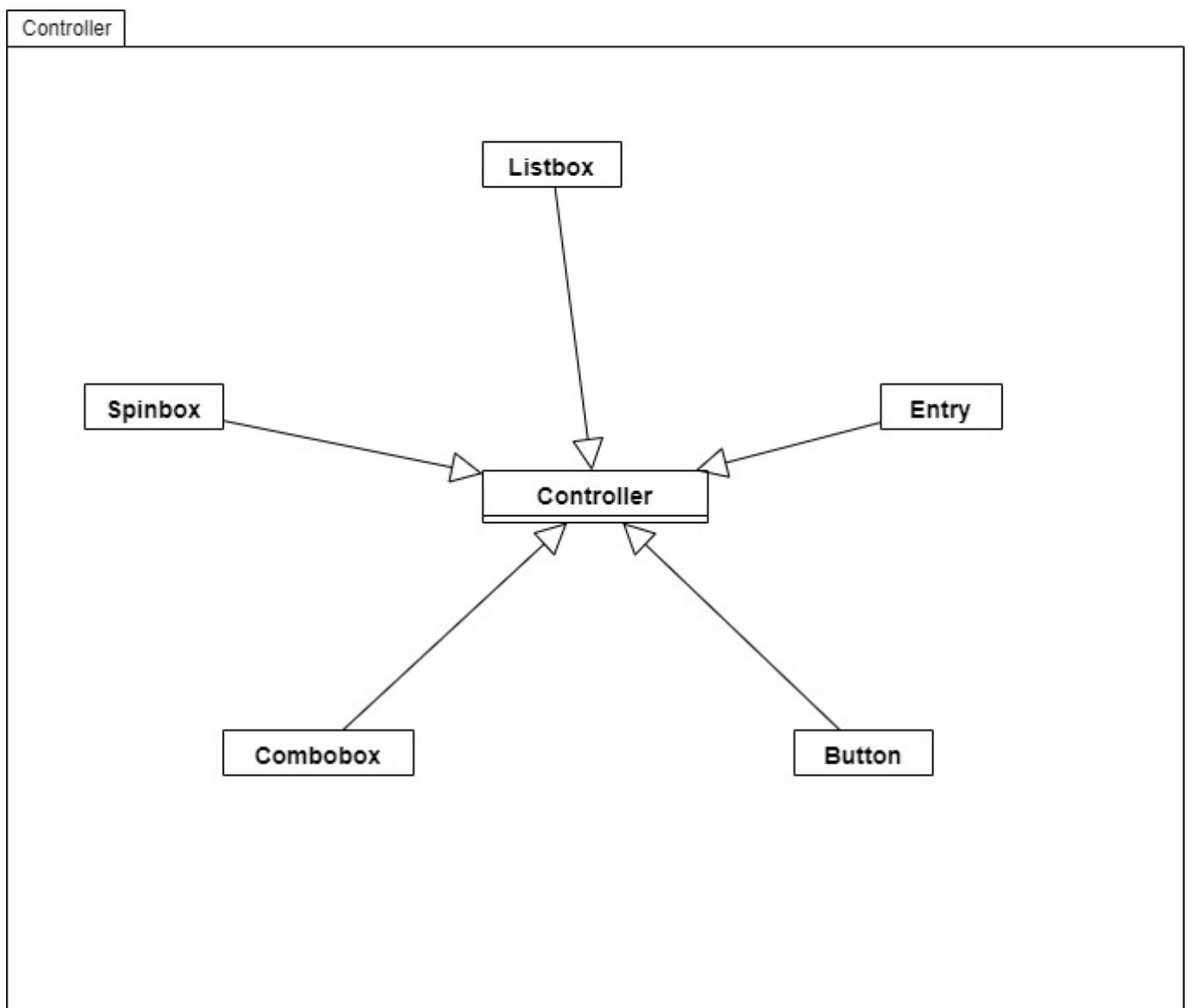
Diagramme du Paquet Model :



Conception UML

L'Axe Statique

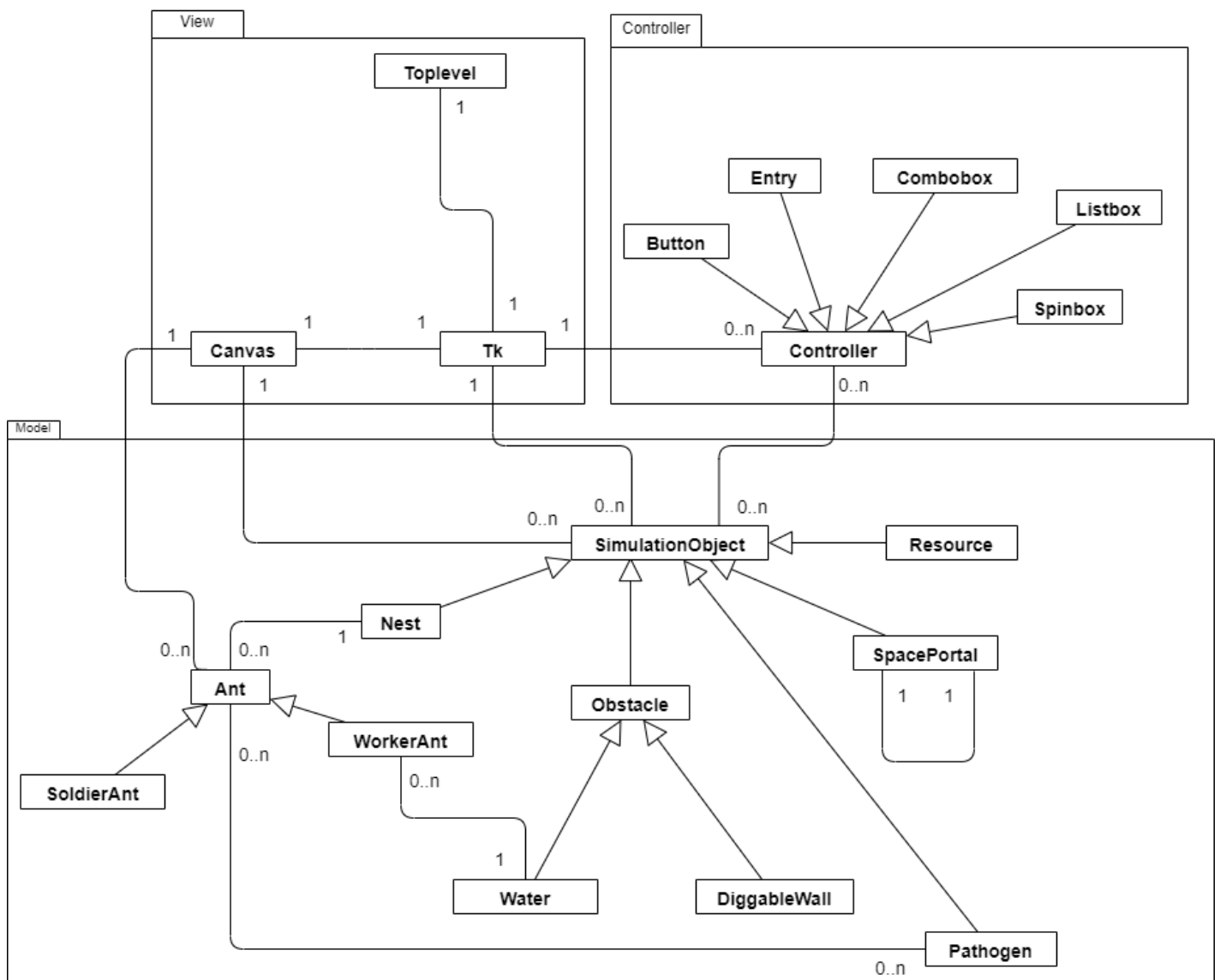
Diagramme du Paquet Controller :



Conception UML

L'Axe Statique

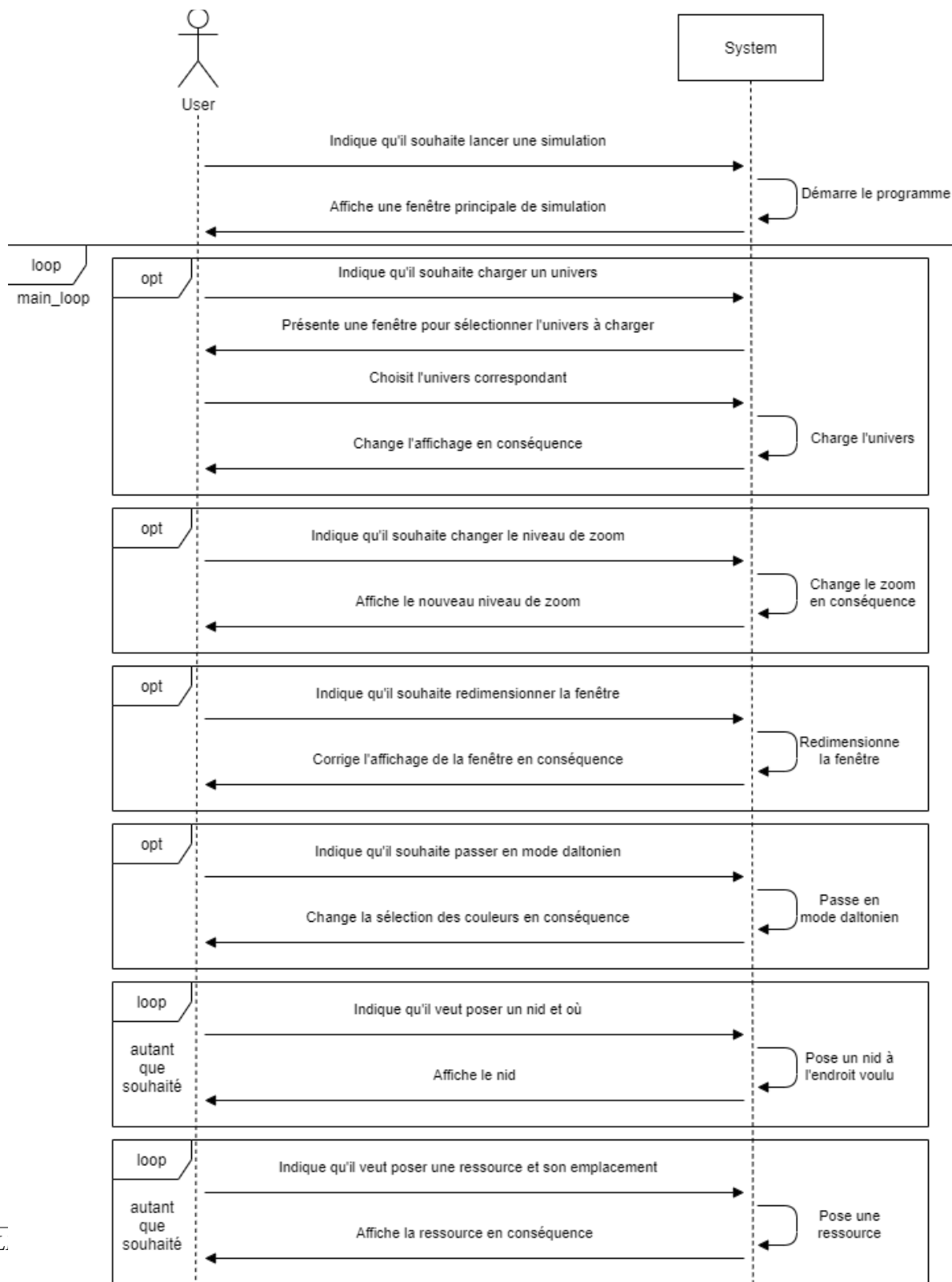
Diagramme de Classes :



Conception UML

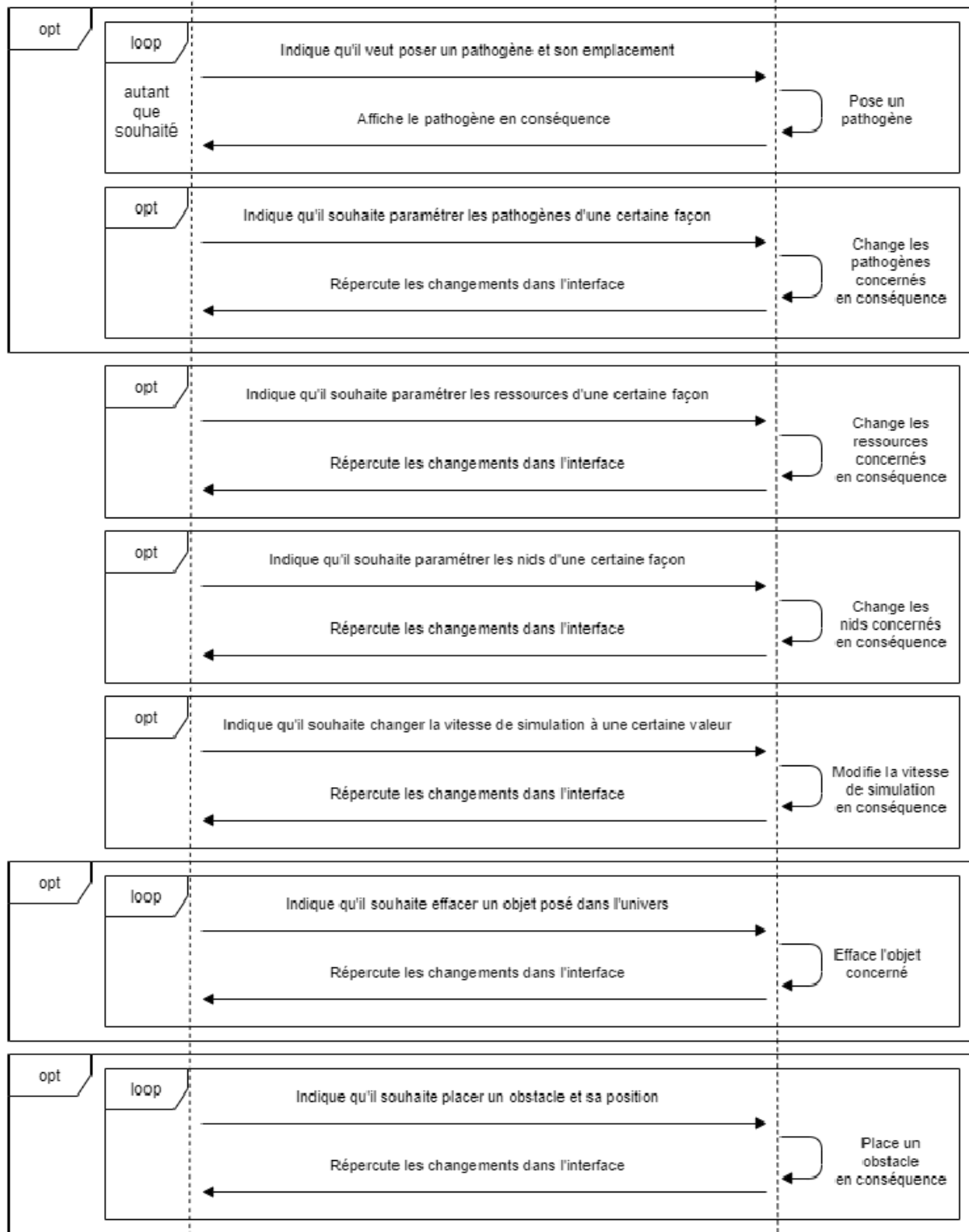
L'Axe Dynamique

Diagramme de Séquence :



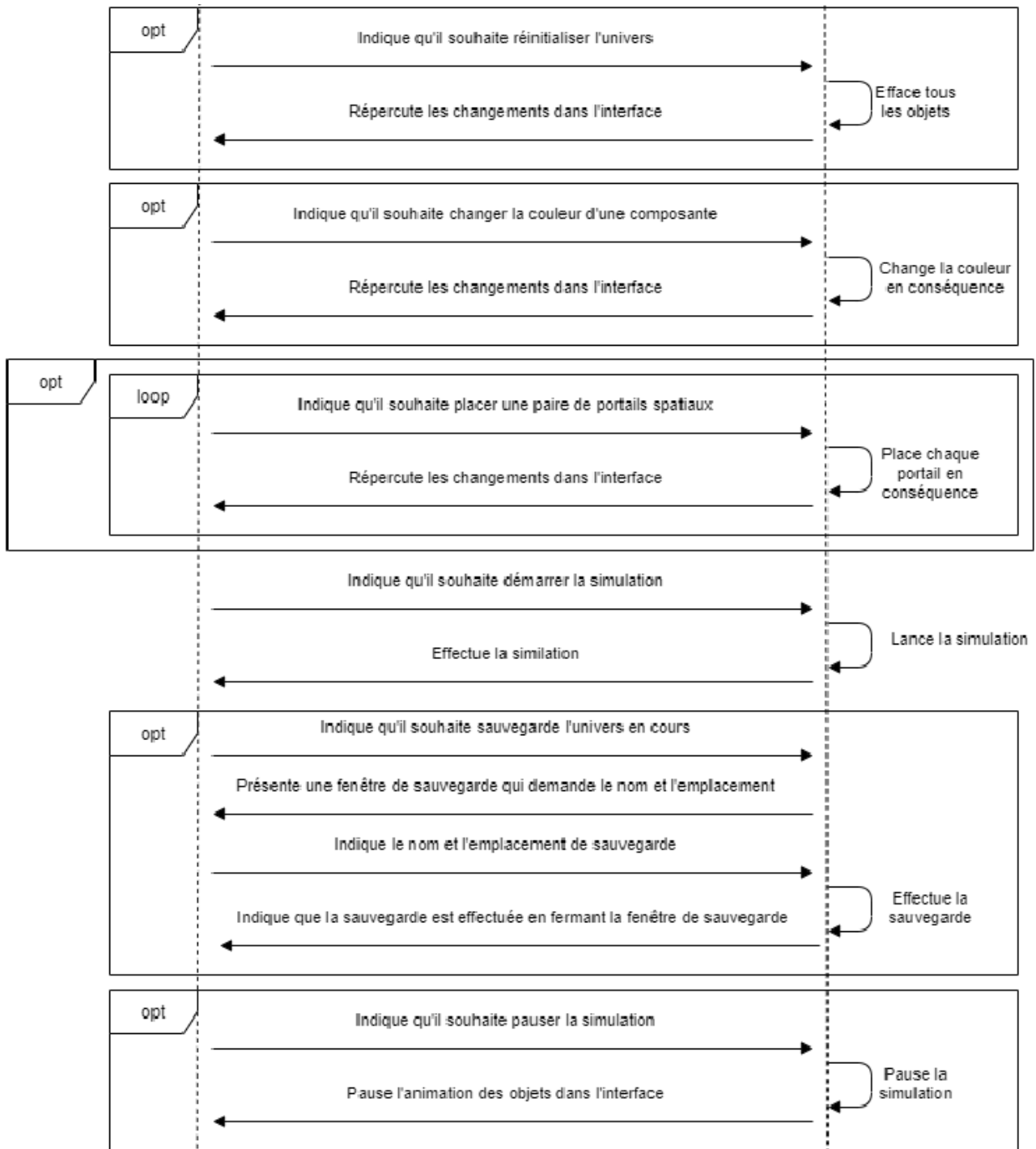
Conception UML

L'Axe Dynamique



Conception UML

L'Axe Dynamique



Manuel Utilisateur

Les Menus

File:

Après avoir cliqué sur le menu « file », donne accès à plusieurs fonctionnalités :

- « New » : supprime la simulation actuelle pour en avoir une vierge.
- « Save » : enregistre la simulation actuelle afin de pouvoir la rouvrir plus tard.
- « Open » : remet en place une simulation préalablement enregistré.

Option:

Après avoir cliqué sur le menu «Option», donne accès à plusieurs fonctionnalités :

- « Resize » : donne la possibilité de modifier la largeur (« Width ») et la hauteur (« Height ») de la fenêtre. Cliquer sur le bouton « Ok » pour confirmer les nouvelles dimensions ou bien « Annuler » pour ne rien changer.

Manuel Utilisateur

Les Boutons

Nest :

Après avoir cliqué sur le bouton « Nest », le clic gauche placera un nid.

Resource :

Après avoir cliqué sur le bouton «Resource», le clic gauche placera une ressource.

Pathogene:

Après avoir cliqué sur le bouton «Pathogene», le clic gauche placera un pathogène.

Obstacle:

Après avoir cliqué sur le bouton «Obstacle», le clic gauche placera un obstacle jusqu'au relâchement de la souris.

Portal:

Après avoir cliqué sur le bouton «Portal», le clic gauche placera un portail, les portails marchent par paire.

Select:

Après avoir cliqué sur le bouton «Select», le clic gauche affichera les caractéristiques de l'élément ciblé. (voir modifier caractéristique)

Effacer:

Après avoir cliqué sur le bouton «Effacer», le clic gauche effacera l'élément ciblé.

Zoom:

Après avoir cliqué sur le bouton «Zoom», le clic gauche agrandit la simulation sur la zone ciblée.

Manuel Utilisateur

Les Boutons

Dezoom:

Après avoir cliqué sur le bouton «Dezoom», le clic gauche rétrécit la simulation sur la zone ciblée.

Start:

Après avoir cliqué sur le bouton «Start», la simulation démarre.

Pause:

Après avoir cliqué sur le bouton «Pause», la simulation se met en attente.

Les paramètres par défaut:

Après avoir cliqué sur les boutons en forme de flèche, donne la possibilité de modifier les valeurs par défaut des différents éléments placés à leur gauche.

La vitesse:

Modifier la valeur sur l'échelle de vitesse changera immédiatement la vitesse d'exécution de la simulation. Plus la valeur est grande, plus la vitesse est élevée.

Manuel Utilisateur

La Simulation

Le but de la simulation est de reproduire le comportement des fourmis dans un milieu créé.

Vous avez la possibilité de placer de multiples nids, ressources, pathogènes, obstacles ou encore portails afin de constater comment cela va influencer leur comportement.

Pour cela les fourmis suivent des mouvements aléatoires tout en étant influencées par les phéromones. Les phéromones sont laissées par les fourmis lorsqu'elles se déplacent. Plus il y a de phéromones, plus la fourmi est susceptible de se diriger dans cette direction.

Les fourmis essaient de trouver des ressources, lorsqu'elles en trouvent, elles ramènent les ressources récoltées à leur nid.

Cependant, elles ne peuvent pas différencier les ressources, des pathogènes, elles peuvent donc s'infecter en récoltant des pathogènes et contaminer d'autres fourmis.

Les fourmis ont aussi de l'endurance, c'est à dire qu'en se déplaçant, si une fourmi n'a plus d'endurance, elle retourne au nid pour regagner de l'endurance puis repart pour de nouvelles explorations.

Conclusion

Pour conclure, ce projet nous a permis de nous rendre compte que la conception UML d'un projet permet de mieux appréhender les fonctionnalités voulus ainsi que ce qu'elles requièrent.

Passer par cette conception UML, nous a permis d'éviter certains problèmes et de nous rendre compte d'incohérences avant le codage de l'application.

Cela a aussi permis à notre groupe d'avoir une idée claire de ce vers quoi on se dirigeait.

En effet, malgré les conditions qui nous ont contraint à travailler tout du long à distance, nous avons réussi à mener à bien ce projet ensemble en utilisant des moyens de partage d'espace de travail et à se répartir les tâches sans incompréhension.

Ainsi, malgré les conditions particulières liées au confinement, nous avons réussi à travailler en groupe en nous adaptant à l'emploi du temps de chacun. Nous avons essayé de tenir compte des idées de tous, chacun a fait des concessions sur ses propres idées. Nous avons dû apprendre à mieux exprimer nos points de vue. En effet, quelque chose d'évident pour quelqu'un ne l'ai pas pour tous. Nous avons appris la gestion du travail, tant en terme de répartition des tâches, qu'en terme de procédure d'action.

Cette expérience nous aura donc permis de mesurer l'importance de la conception UML à la fois pour avoir une vision claire du projet, et pour économiser du temps en programmation.

Nous avons donc éprouver le travail de groupe dans tout ce qu'il peut apporter ainsi que dans les erreurs à éviter pour le faciliter.

Annexe

Les Exigences

1. Exigences d'interface:

EXG-IHM-0010: L'utilisateur doit pouvoir placer un nid

EXG-IHM-0020: Lorsqu'un nid est placé, le système doit attribuer une couleur correspondant à une faction au nid en question.

EXG-IHM-0030: L'utilisateur doit pouvoir placer une ressource

EXG-IHM-0040: L'utilisateur doit pouvoir placer un obstacle

EXG-IHM-0050: L'utilisateur doit pouvoir lancer une simulation

EXG-IHM-0060: L'utilisateur doit pouvoir passer en mode daltonien

EXG-IHM-0070: L'utilisateur doit pouvoir sauvegarder et charger l'univers

EXG-IHM-0080: L'utilisateur doit pouvoir poser des portails spatiaux

EXG-IHM-0090: L'utilisateur doit pouvoir introduire un pathogène dans l'univers

EXG-IHM-0100: L'utilisateur doit pouvoir poser différents types d'obstacles, y compris de l'eau

2. Exigences de paramétrisation:

EXG-PAR-0010: L'utilisateur doit pouvoir paramétrer les ressources

EXG-PAR-0020: L'utilisateur doit pouvoir paramétrer les obstacles

EXG-PAR-0030: L'utilisateur doit pouvoir paramétrer le nombre de fourmis dans chacun des nids

EXG-PAR-0040: L'utilisateur doit pouvoir régler la vitesse

EXG-PAR-0050: L'utilisateur doit pouvoir changer la taille du rayon dans lequel les fourmis combattent lors d'une rencontre hostile

EXG-PAR-0060: L'utilisateur doit pouvoir modifier l'angle du cône d'odorat des fourmis

EXG-PAR-0070: L'utilisateur doit pouvoir paramétrer divers aspects de la production de fourmis par les nids tels que l'intervalle de temps avant chaque portée, le nombre de fourmis par portée, le coût en ressource de chaque fourmis, etc.

EXG-PAR-0080: L'utilisateur doit pouvoir paramétrer la proportion de fourmis soldat produites pour chaque nid

3. Exigences de robustesse:

EXG-ROB-0010: Lorsqu'un nid est placé, le système doit placer un nombre de fourmis correspondant différent de 0.

4. Exigences fonctionnelles:

EXG-FCT-0010: Le système doit attribuer une endurance à chaque particule

EXG-FCT-0020: Lorsque l'endurance d'une particule atteint 0, le système doit forcer la particule à retourner au nid

EXG-FCT-0030: Lorsqu'une particule retourne au nid, le système doit restaurer son endurance

EXG-FCT-0040: Lorsqu'une particule rencontre un obstacle, le système doit forcer la particule à le contourner

EXG-FCT-0050: Lorsqu'une particule rencontre une ressource, le système doit la forcer à retourner au nid avec une partie de la ressource et laisser des phéromones au sol sur le chemin du retour

EXG-FCT-0060: Lorsqu'une particule rencontre des phéromones, le système doit la forcer à préférentiellement suivre les phéromones

EXG-FCT-0070: Lorsqu'une particule rencontre une particule du même nid, le système détermine si seulement une d'entre elle transporte une ressource, si c'est le cas, l'autre particule doit céder la place, sinon, dans tous les autres cas, un choix aléatoire est fait sur laquelle cède la place à l'autre

EXG-FCT-0080: Lorsqu'une particule rencontre une particule d'un autre nid, le système doit compter le nombre de particules de chaque nid dans un rayon autour arbitraire. Ces particules doivent combattre

EXG-FCT-0090: Le système ne doit accorder qu'un angle de déplacement réduit à chaque particule

EXG-FCT-0100: Le système doit accorder à chaque particule un cône d'odorat mais pas de vision

EXG-FCT-0110: Le système doit garder les phéromones en mémoire de façon discrétisée

EXG-FCT-0120: Lorsqu'une particule rencontre le bord de l'écran, le système doit la faire rebondir

EXG-FCT-0130: Lorsque les fourmis rencontrent de l'eau, le système doit leur accorder la possibilité de se sacrifier pour construire un pont

EXG-FCT-0140: Lorsque les fourmis rencontrent un mur creusable sur le retour d'une ressource, le système doit les forcer à creuser un tunnel vers le nid (éprouve l'endurance). Si elles n'ont plus d'endurance elles ne peuvent plus creuser

EXG-FCT-0150: Lorsqu'un certain laps de temps est écoulé, le système doit vérifier pour chacun des nids si il possède assez de ressource pour produire une nouvelle fourmi. Si c'est le cas les ressources sont consommées et une nouvelle fourmi est produite

EXG-FCT-0160: Lorsqu'une fourmi est produite par un nid, le système détermine selon ses paramètres si il s'agit d'une fourmi ouvrière ou d'une fourmi soldat