

- Introduction

Dans le cadre du troisième laboratoire en GPA434, nous avons reçu le mandat de créer une application graphique qui peut simuler un écosystème ayant au moins 3 entités. Il est important que celui-ci doit être représenté par un cycle complet.

- Présentation générale du devis

Nous avons travaillé sur l'implémentation d'un écosystème contenant des dinosaures. Plus précisément, nous avons comme entités: arbres, brotons saures, pterodactyles, trex.

Nous allons donc simuler un écosystème préhistorique.

- Présentation spécifique du devis

Les arbres représentent des entités statiques qui donne de la nourriture lorsque consommés par des herbivores. Les entités dynamiques sont caractérisés par une classification entre herbivores et carnivores. Les carnivores sont des prédateurs qui mangent des herbivores alors que ceux-ci mangent des arbres. Lorsqu'un animal meurt, la carcasse crée de l'engrais pour la terre qui fait pousser un arbre.

- Présentation des paramètres

Nous avons beaucoup introduit de paramètres non configurables qui sont initialisés lors d'appel de constructeur avec des paramètres optionnels. Nous avons gardé un paramètre initial de la taille de population qui crée aléatoirement une quantité de chaque entité.

Dans les constructeurs avons aussi la position initiale (qui est décidée aléatoirement), la couleur, l'âge initial ainsi que la vitesse.

- UML

Voir le fichier attaché en annexe.

- Captures d'écran

Voir les photos attachées en annexe.

- Résultats et discussions

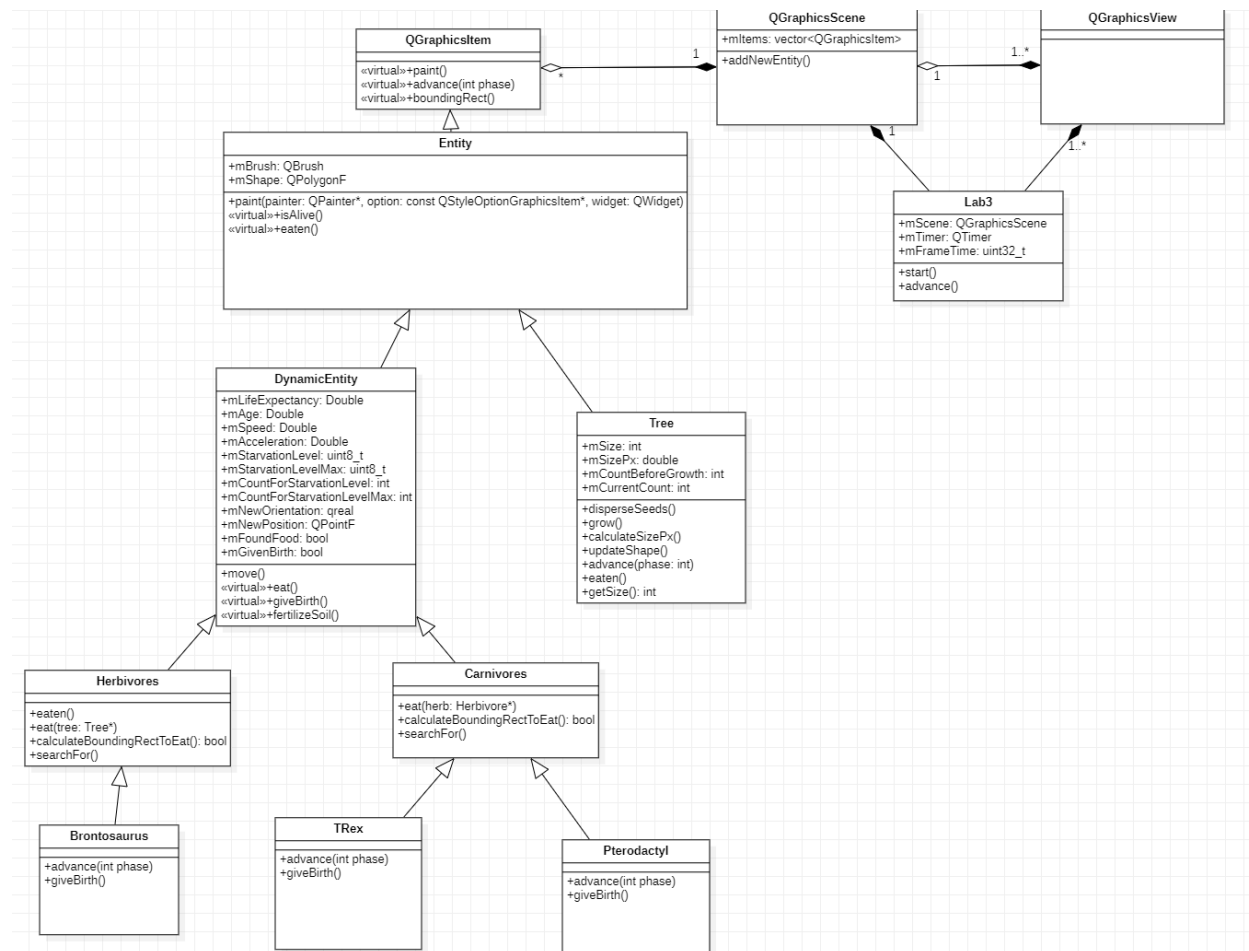
Nous croyons que la conception de logiciels avec UML permet de faciliter énormément le développement de logiciels à plus grande échelle. À l'aide d'un départ plus lent, nous pouvons mieux prévoir la hiérarchie du logiciel.

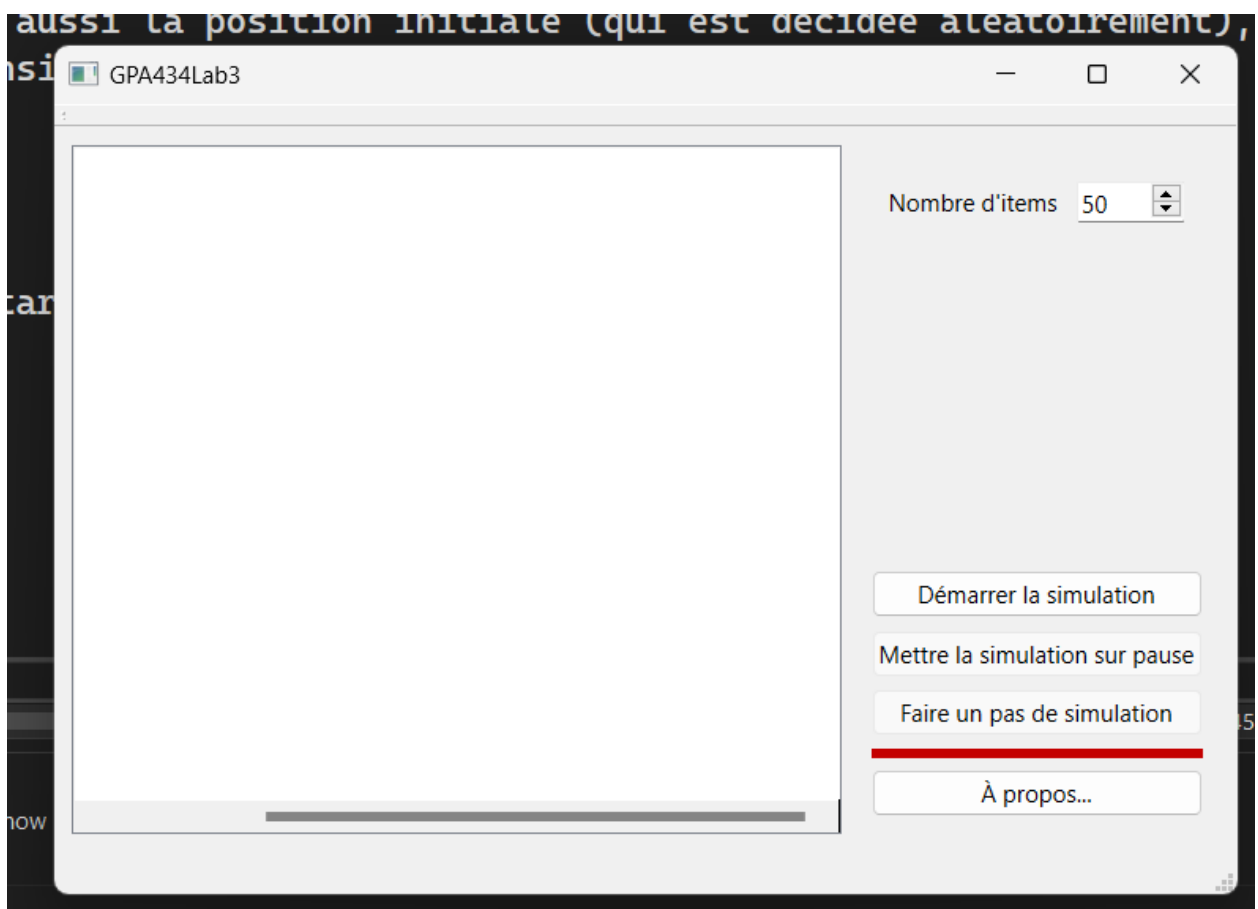
Puisque ce projet est notre premier qui nous fait concevoir du UML, il a été plus difficile que prévu de concevoir comment chaque méthode allait s'appeler. Malgré notre bon départ,

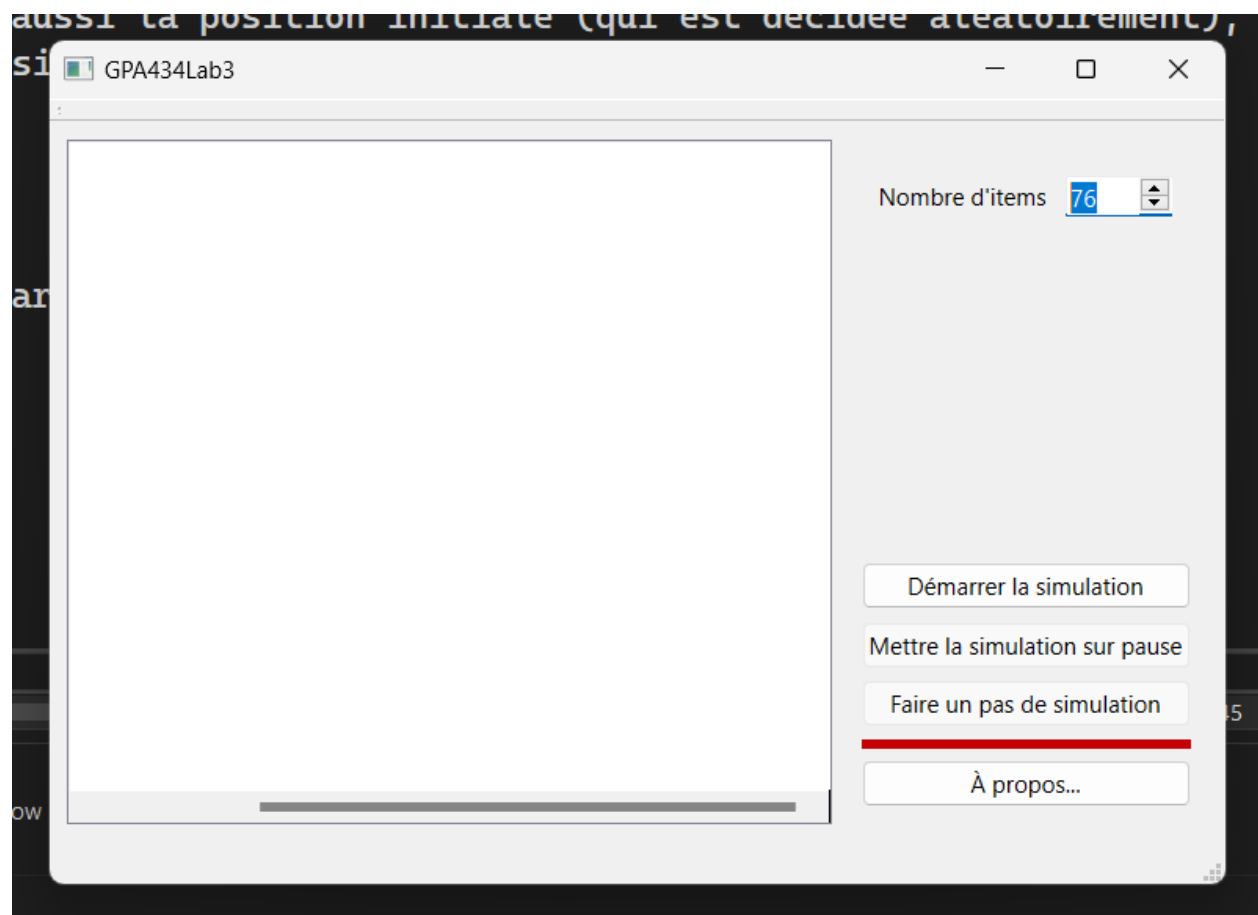
nous avons beaucoup modifié le UML initial au cours de notre laboratoire en retouchant les fonctions lorsque nécessaire.

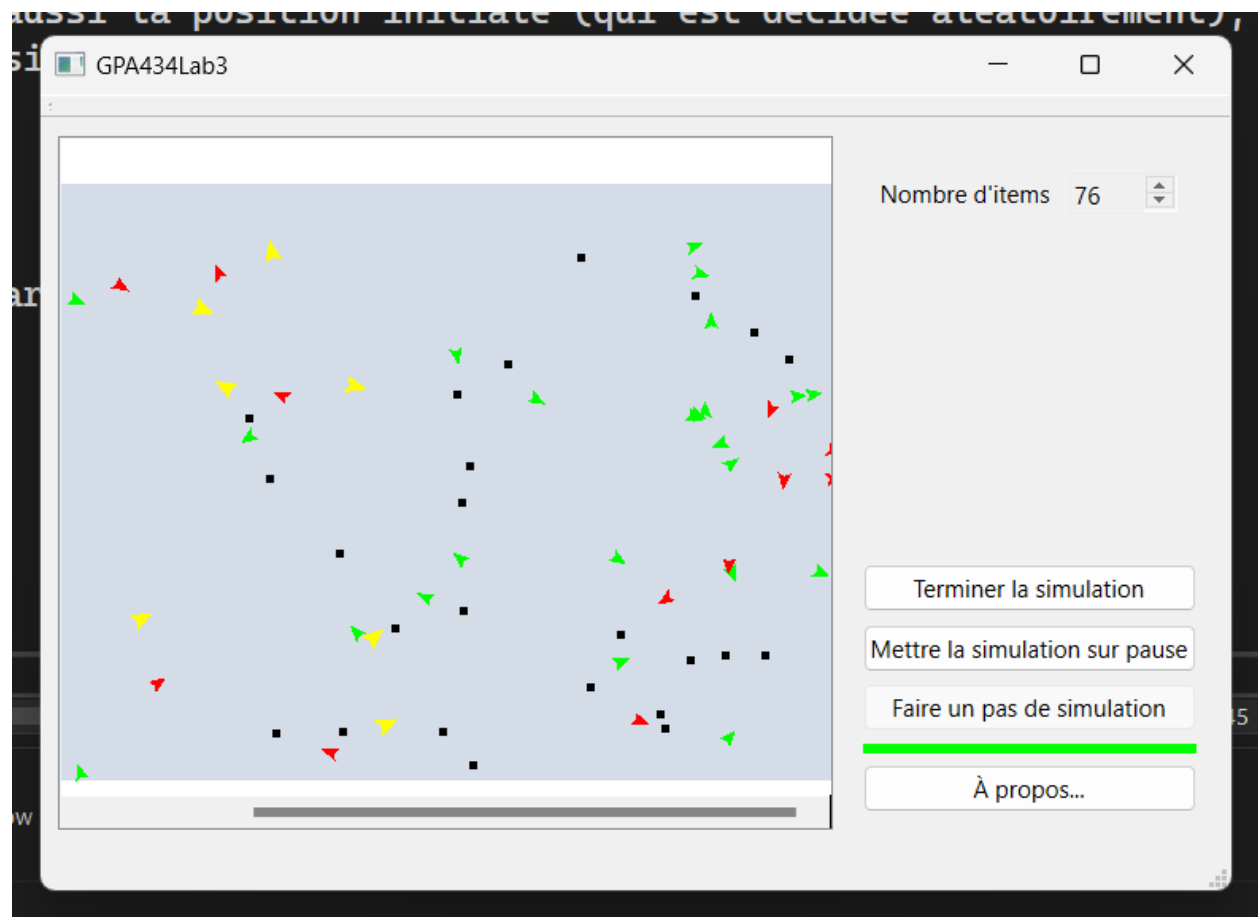
La collision entre les élément ainsi que la modification de l'angle de chaque élément lors de la détection de nourriture a été plutôt difficile.

Nous croyons que notre solution est de bonne qualité et puisque nous avons suivi les paradigmes orientés objets, nous aurons donc une faciliter à travailler sur notre projet dans le futur et a ajouter certaines classes.









aussi la position initiale (qui est decidee aleatoirement),

