BRUNET-MANQUAT Ivan

Modélisation UML de l'application

Diagrammes de classes

Etant donné le nombre conséquent de classes contenues dans ce projet et les contraintes d'affichage, la modélisation UML de l'application est réalisée en 3 diagrammes de classes :

1/ Un diagramme ne représentant que la partie «model» de l'application.

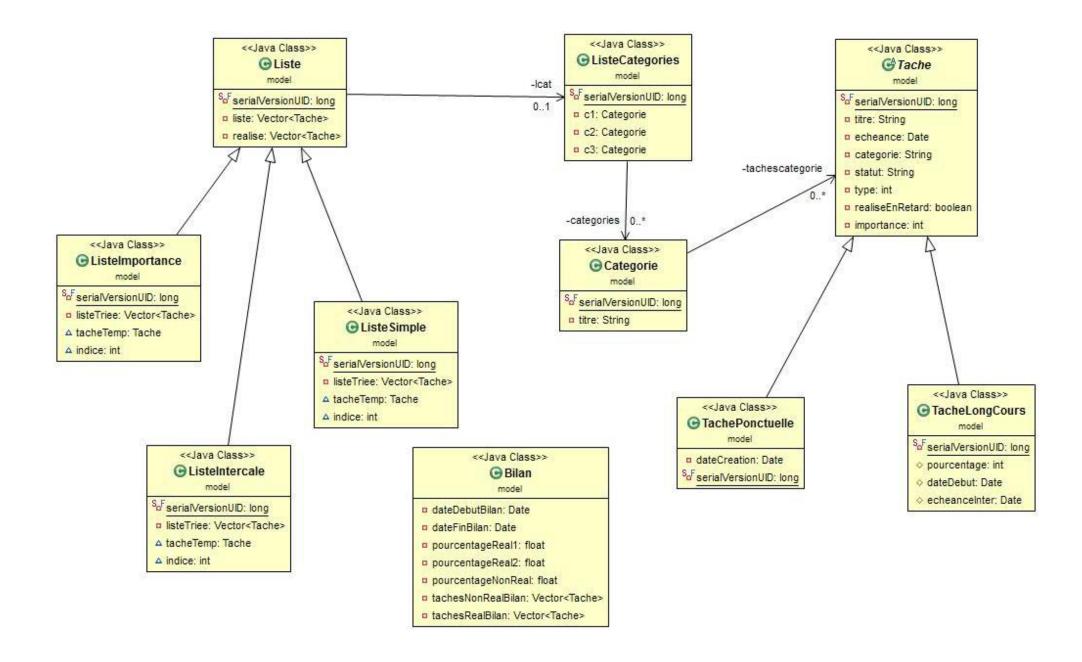
Pour ce diagramme, l'affichage des méthodes a été masqué.

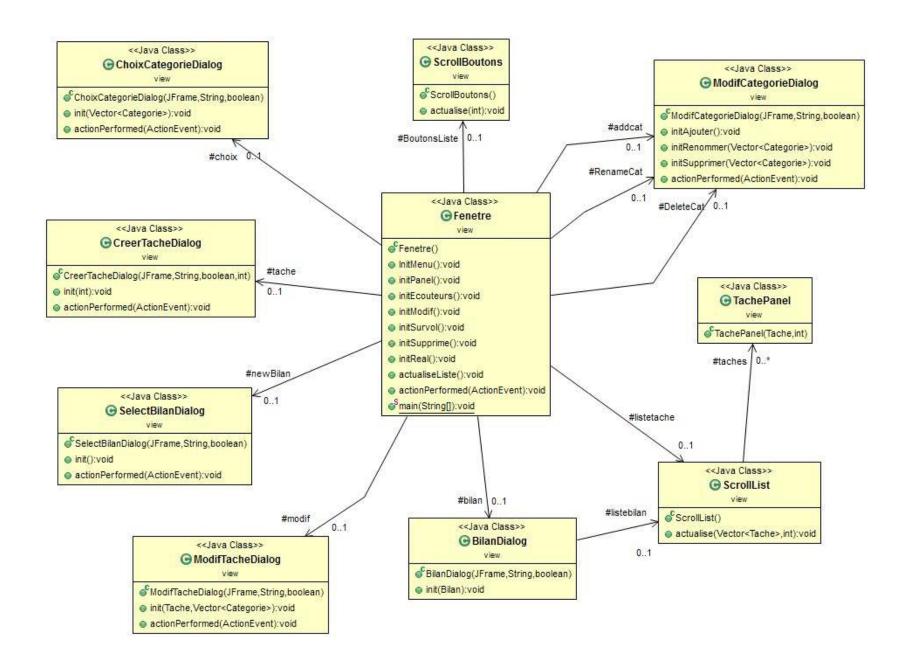
2/Un diagramme ne représentant que la partie «view» de l'application.

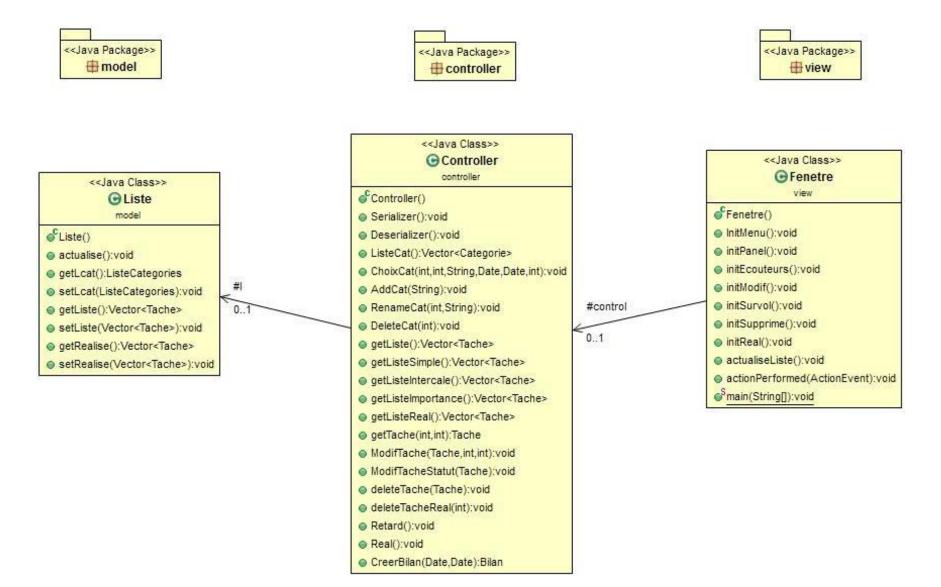
Pour ce diagramme, l'affichage des attributs a été masqué.

3/ Un diagramme représentant l'interaction entre la partie « model » et la partie «view» par le biais de la partie «controller»

Pour ce diagramme, l'affichage des attributs a été masqué.







Comme le montre ces diagrammes de classes, la partie métier de l'application a été dissocier de la partie graphique, grâce à un model MVC. La mise en place d'un model MVC n'ayant pas été vu en cours, et les différent exemples d'implémentations possibles que l'on peut trouver sur internet étant nombreux, je me dois d'expliquer mes choix de conception.

Il y a 3 packages : model, view et controller qui contiennent respectivement les classes leur correspondant.

J'ai fait le choix que la partie vue (contrôlée par la classe fenêtre) contienne l'instance de la partie « controller » qui ellemême contient l'instance de la partie « model ».

La partie « controller » fait le lien entre la vue et le model. Lorsque l'utilisateur interagit avec la fenêtre (ex : bouton créer une tâche), son action est captée par des « Actionlistener », la fenêtre appelle alors une méthode du « Controller » qui, après traitement, appelle lui-même une méthode du model afin d'actualiser les données (ex : données d'une tâche saisies dans la partie vue). Ces deux méthodes retournent alors les éléments nécessaires à la vue pour s'actualiser à son tour. Cette méthode de conception permet une meilleure visualisation globale du projet. De plus, cette dissociation favorise une détection rapide des bugs.

Diagramme des machines à états pour les tâches

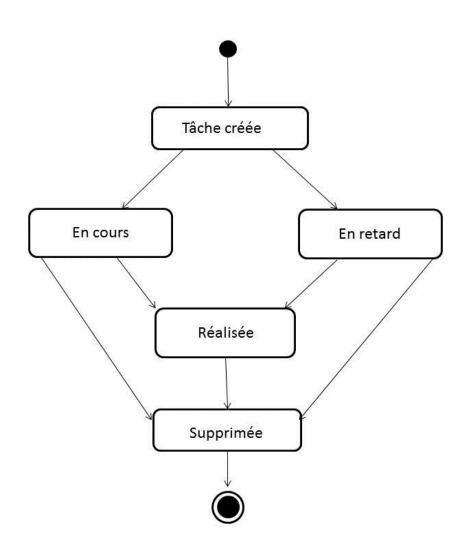


Diagramme de séquence système (interaction : Création de tâche)

