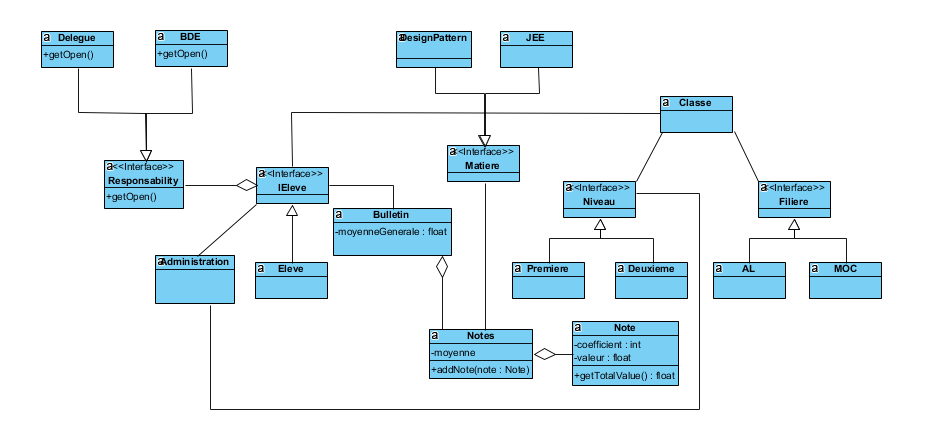
# Design Patterns

Nous avons choisi comme sujet le développement d’une application permettant de gérer une école avec leurs élèves, notes, classes etc.

Voici le diagramme de classe :

Pour représenter une classe il faut un objet composé du niveau (4er année, etc.) et de de la filière (Al, MOC) Nous avons utilisé le pattern Builder car l permet de

(fatory method a la base ??? )

Nous avons également remarqué que pour ne pas avoir à recréer plusieurs instances de filière ou de niveau lorsque l’on souhaite créer d’autre classe le pattern FlyWeigth était parfaitement adapter à cette situation.

Pour le calcul des moyennes lors de l’ajouter et la suppression d’une note nous avions dans un premier temps pensé a utilisé le pattern mémento. Cependant memento ne permet pas la modification. C’est pourquoi nous avons finalement utiliser le pattern observer afin de mettre à jour la moyenne dès qu’une note est modifié, ajouté ou supprimé.

Il y a une fonctionnalité qui permet de savoir si un élève a validé son année scolaire. Cette méthode prend en compte plusieurs choses en compte tel que les point Open, les notes obtenus, le score du TOEIC obtenu … Pour résoudre cela nous utilisons le pattern states car on peut changer le comportement de l’objet quand son état change.

Nous avions besoin d’une méthode pour donner des point Open d’un élève. Ces points étaient donner pour différente raison comme par exemple être délégué, être au BDE ou dans une association. De plus, chaque activité donner un nombre de point propre et il est possible de rajouter des activités par la suite. C’est pourquoi nous avons utilisé le pattern Bridge cela nous permet de mieux structurer notre code ainsi que de rester ouvert.

VISITOR ( a la base décorator )