ISTIC

Rapport TP AOC

Métronome

Achraf Ibn Mrabet & Ismail Mellali

10/01/2017

M2 MIAGE (SIAD)

Ce document présente le rapport de notre travail qui consistait à créer un métronome en déployant les différents Design Pattern.

Sommaire

I. Introduction

Le module d'AOC est une opportunité pour les étudiants de découvrir les design patterns et les manipuler. Ces derniers sont la base de quelque Framework (.NET).

Un métronome est un instrument donnant un signal audible ou visuel permettant d'indiquer un tempo, vitesse à laquelle doit être jouée une musique. Il est surtout utilisé dans l'étude d'une partition, la mise en place d'une interprétation ou la recherche du minutage (*timing*) d'une œuvre musicale. Source: Wikipedia

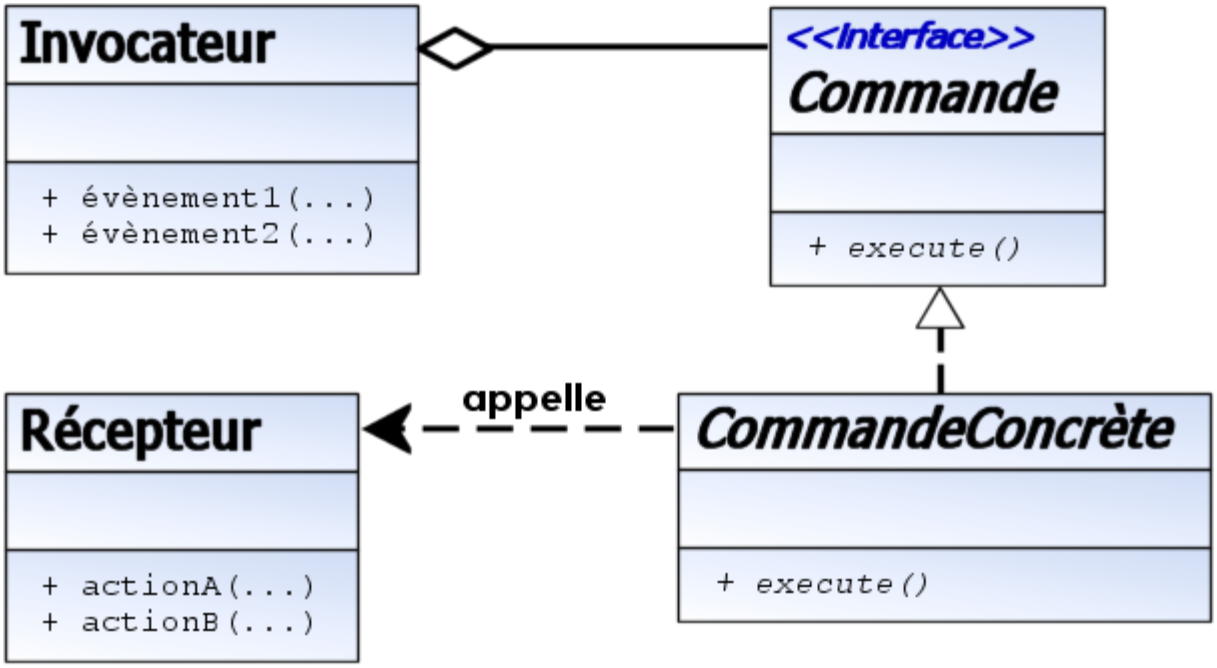
Pour notre TP qui consiste à concevoir une application de type métronome possédant les caractéristiques suivantes :

- Un clavier de commande muni de touches de contrôle ;
- Un curseur de réglage du tempo (nombre de temps par minute) ;
- Un afficheur de tempo ;
- Deux LEDs, l'une marquant chaque temps par un éclair bref, l'autre marquant le premier temps de chaque mesure par un éclair bref ;
- Un haut-parleur mettant un clic sonore à chaque temps.

II. Architecture

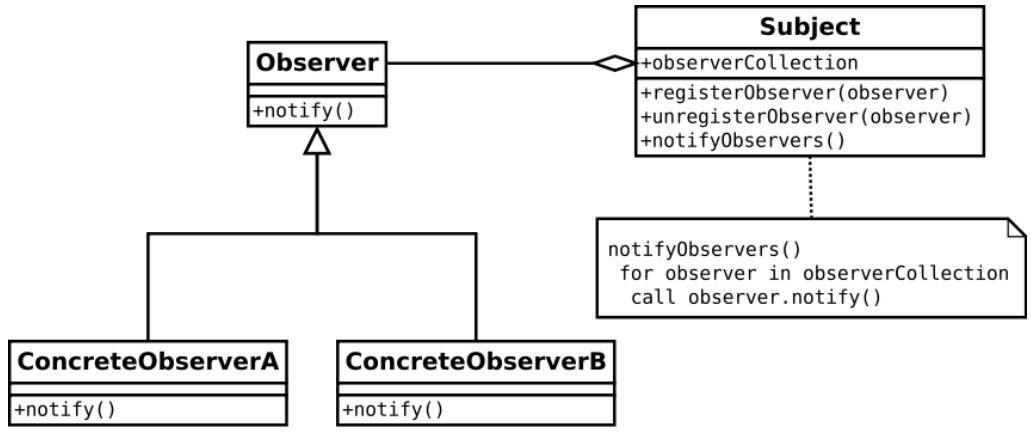
Nous avons utilisé deux design patterns pour cette version :

- **Command :**



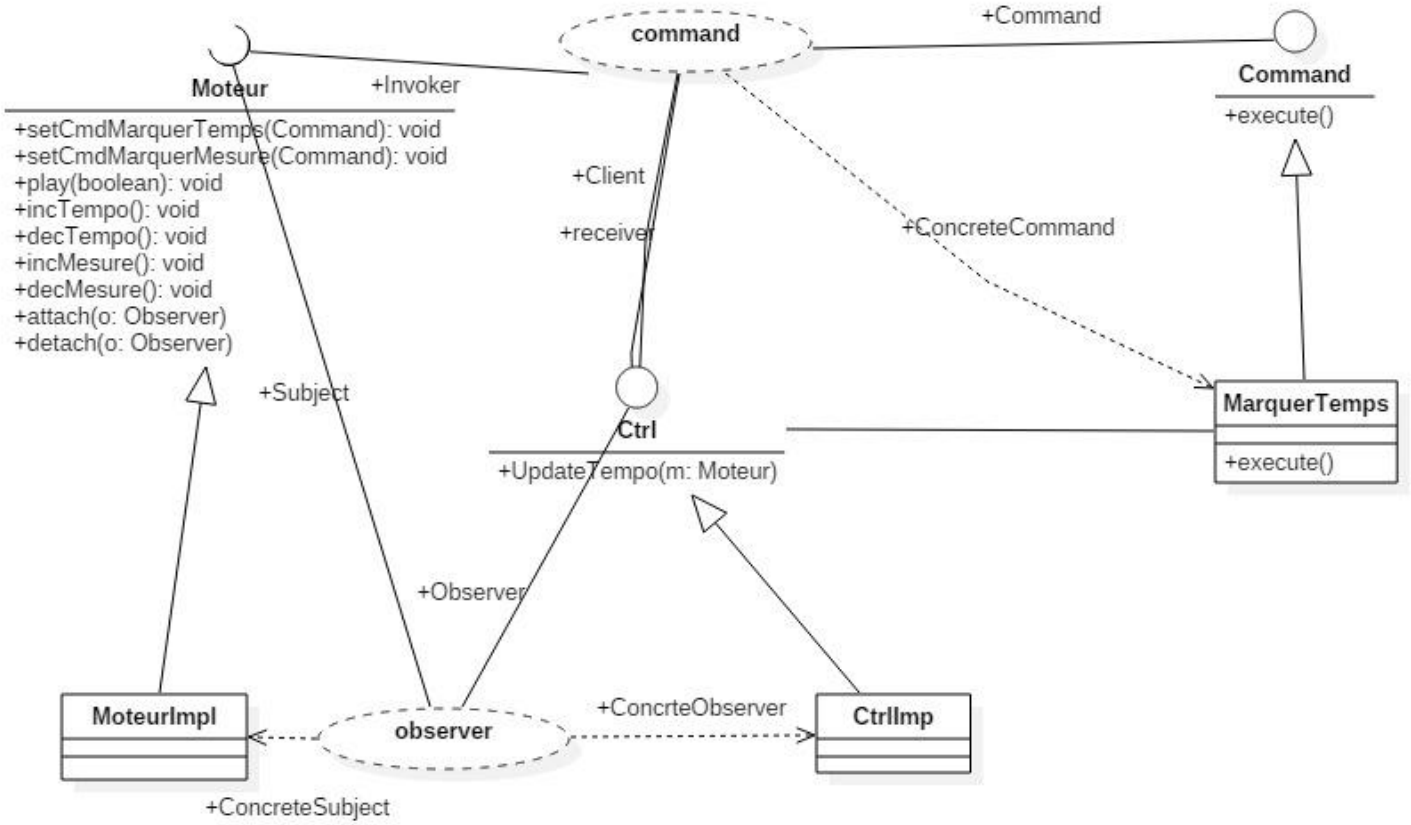
Source: Wikipedia

- **Observer**



Source: Wikipedia

Diagramme UML :



Comme le montre le digramme ci-dessus nous avons déployé le DP Command et Observer, cela pour diminuer les dépendances entre le moteur et le Controller.

Le moteur du métronome est le sujet, il est observé par le Controller pour que ce dernier exécute l'action correspondante avec l'IHM (nous verrons par la suite).

Il existe plusieurs commandes que notre moteur peut invoquer, il fait appelle à la commande correspondante et délègue au Controller d'exécuter l'action final.

Les différentes commandes possibles sont :

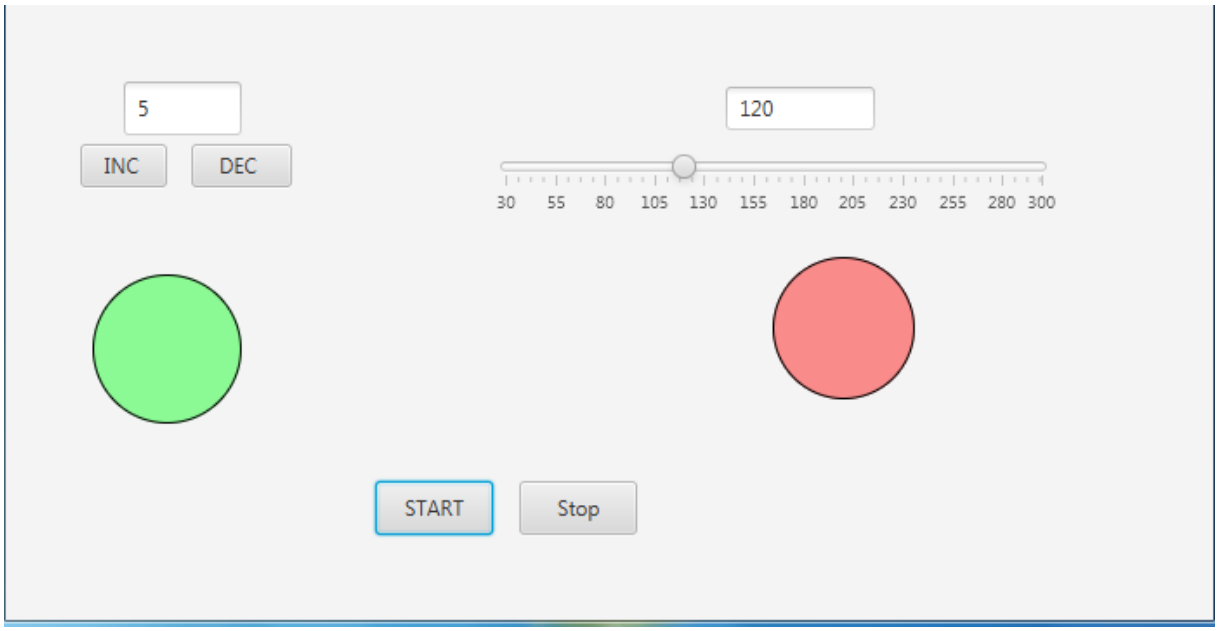
- MarquerMesure : marque la mesure par un son et la LED Rouge.

- MarquerTemps : marque le temps par un son et la LED verte.

Le moteur est le cœur de notre métronome, et connait toutes les différentes valeurs pour calculer le temps correspond afin d'invoquer la commande.

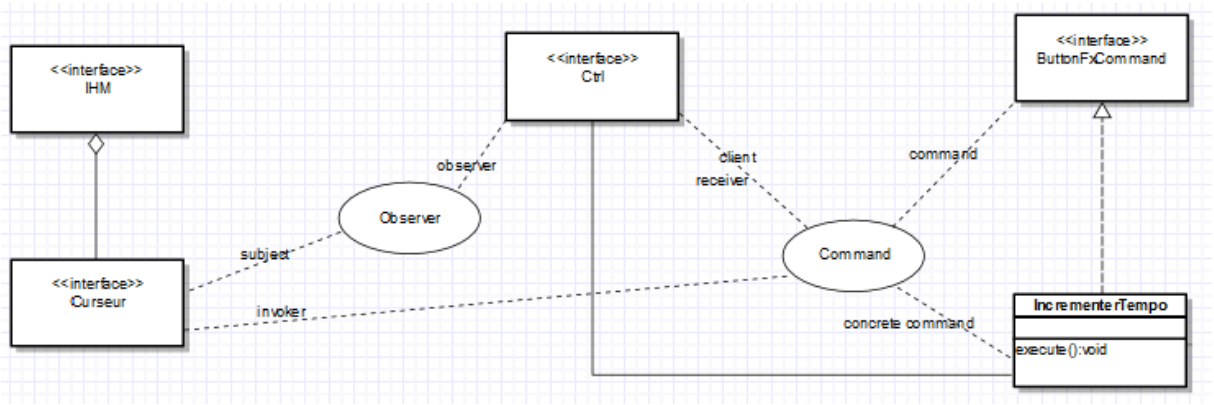
D'un autre côté notre métronome est doté d'une interface graphique contenant tous les boutons et curseur cité ci-dessus.

Prise d'écran de notre interface :



A partir de cette IHM l'utilisateur peut interagir avec l'outil avec la possibilité de lancer arrêter le métronome incrémenter décrémenter le temps et la mesure.

Pour cela nous avons utilisé l'architecture suivante :



L'utilisateur à partir de l'IHM manipule les boutons ou le curseur, une interface CurseurFX ou BouttonFX invoque la commande correspondante et délègue l'exécution de l'action pour le contrôler.

III. Choix technologique

Java :

Notre langage de développement principale, ce dernier orienté objet facilite la manipulation des classes pour nos design pattern.

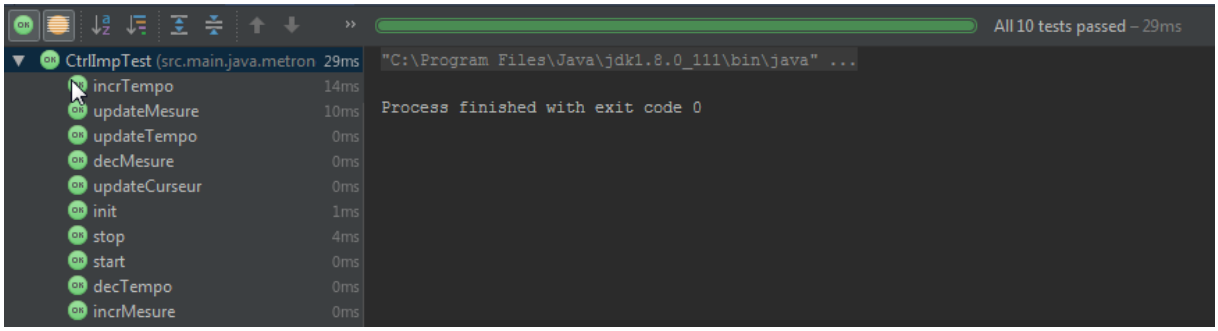
JavaFX & Scene Builder:

Bibliothèque de création d'interface graphique officielle pour le langage JAVA , cette dernière opère avec Scene Builder afin de faciliter la création des interfaces en utilisant le Drag & Drop.

IV. Test

Nous avons réalisé des tests unitaires sur le moteur et le contrôleur du metronome afin de s'assurer le bon fonctionnement des différentes commandes (Incrémenter mesure, décrémenter mesure, marche, arrêt, incrémenter tempo, décrémenter tempo…) .

Test concernant Controller :



Test concernant Moteur :