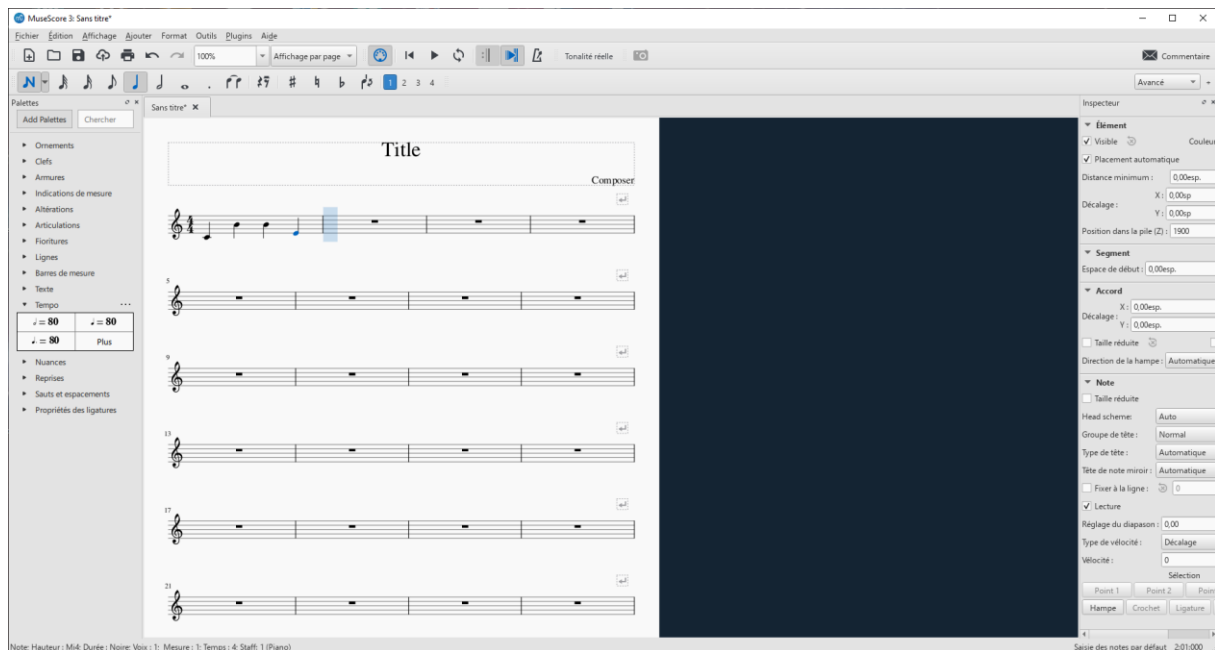


Etude de cas : Conception d'un séquenceur musical



Le séquenceur MuseScore

Besoins principaux :

- On désire réaliser un logiciel permettant de saisir des notes de musique sur des portées.
- La saisie peut se faire note par note mais aussi à l'aide d'un clavier MIDI (synthétiseur, clavier maître, etc..).
- La partition ainsi créée pourra être jouée et sauvegardée dans un fichier.

Besoins secondaires :

- Le clavier MIDI pourra être remplacé par un clavier virtuel.
- On pourra choisir les sons produits lors de la lecture de la partition parmi une banque de sons disponible
- La lecture de la partition pourra être effectuée à différents tempi (tempo par défaut : 120)

Travail demandé

Proposer un diagramme de cas d'utilisation. Pour cela, vous devez identifier les acteurs utilisant votre logiciel et les fonctionnalités de ce dernier.

A partir des descriptions suivantes, établir un diagramme de classes de votre logiciel (bien évidemment, nous n'aborderons qu'une petite partie du logiciel) :

- Une partition est composée d'une ou plusieurs portées
- Une portée est composée de plusieurs mesures
- Une mesure est composée de plusieurs événements musicaux
- Un événement musical peut être soit une note soit un silence
- Une mesure possède une ou plusieurs clefs
- Une mesure peut comporter plusieurs altérations accidentelles

- Une mesure peut appartenir à plusieurs portées (plusieurs voies)
- Une mesure possède un chiffrage
- Une portée possède une armature
- Une note peut être (du plus court au plus long):
 - triple croche
 - triple croche pointée
 - double croche
 - double croche pointée
 - croche
 - croche pointée
 - noire
 - noire pointée
 - blanche
 - blanche pointée
 - ronde
 - ronde pointée
- Un silence peut être (du plus court au plus long) :
 - un 32^{ème} de soupir
 - un 16^{ème} de soupir
 - un 8^{ème} de soupir
 - un 4^{ème} de soupir
 - un demi soupir
 - un soupir
 - pause
 - un baton soupir

Vous utiliserez Enterprise Architect EA pour réaliser ces diagrammes.

A partir d'EA, générer le squelette de votre application : vous devriez obtenir que des .h et des .cpp

Attention : par défaut lorsqu'on ajoute une classe, EA considère que c'est une classe Java. Il faut les basculer en C++

Il faudra bien réfléchir aux visibilité et multiplicités des attributs.

Rappels : La visibilité d'un attribut est positionnée par les mots-clefs *public (+)*, *private (-)* ou *protected (#)*

La multiplicité est la quantité d'objets référencés par un attribut. Par exemple, une mesure possède plusieurs notes donc il y a aura un attribut qui indiquera une association multiple vers la classe *Note*. Cette association multiple sera spécifiée par *0..** au bout de l'association.