

Baillargeon, Samuel
111183612

Gélinas, Émile
111183587

Mercier Aubin, Alexandre
111180756

Venables, Alexandre
111184382

Génie logiciel orienté objet
GLO-2004, NRC : 89638 (sect. A)

Gaudrophone

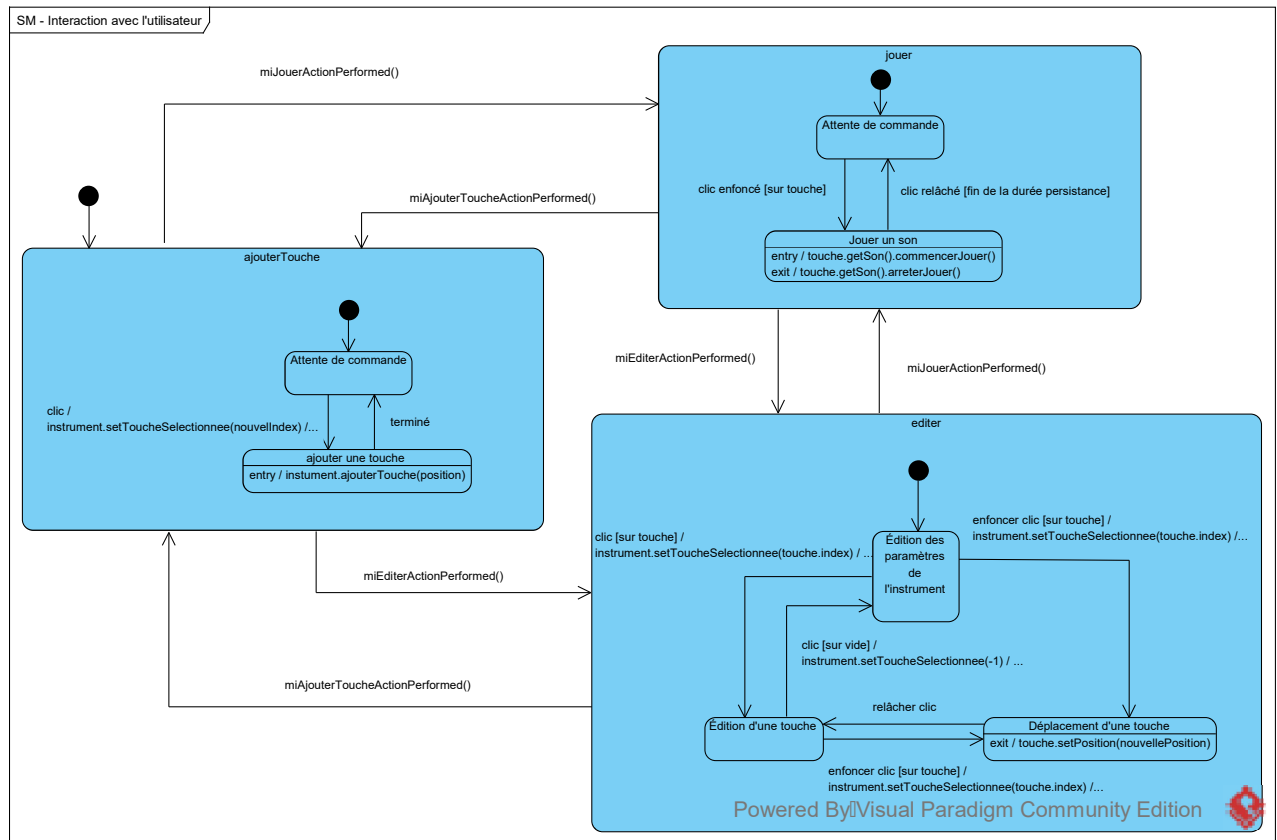
Travail présenté à
Jonathan Gaudreault

Faculté de science et génie
Université Laval
A17

Table des matières

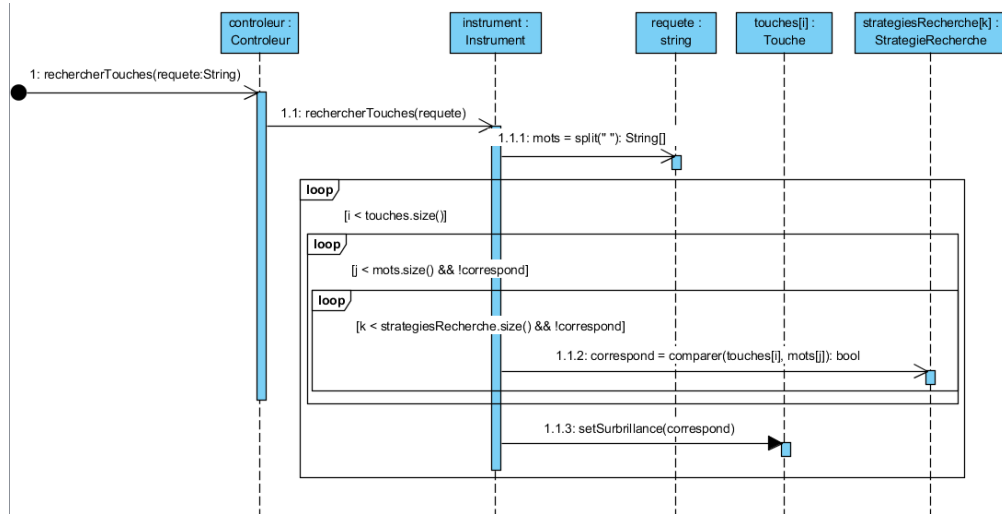
1	Diagramme d'états	2
2	Diagrammes de séquence	3
2.1	Rechercher touches	3
2.1.1	Description	3
2.2	Stratégie Chemin	4
2.2.1	Description	4
2.3	Stratégie Couleur	5
2.3.1	Description	5
2.4	Stratégie Forme	6
2.4.1	Description	6
2.5	Stratégie Note et Octave	7
2.5.1	Description	7
3	Diagramme de classe de conception	8
4	Contribution des membres	9

1 Diagramme d'états



2 Diagrammes de séquence

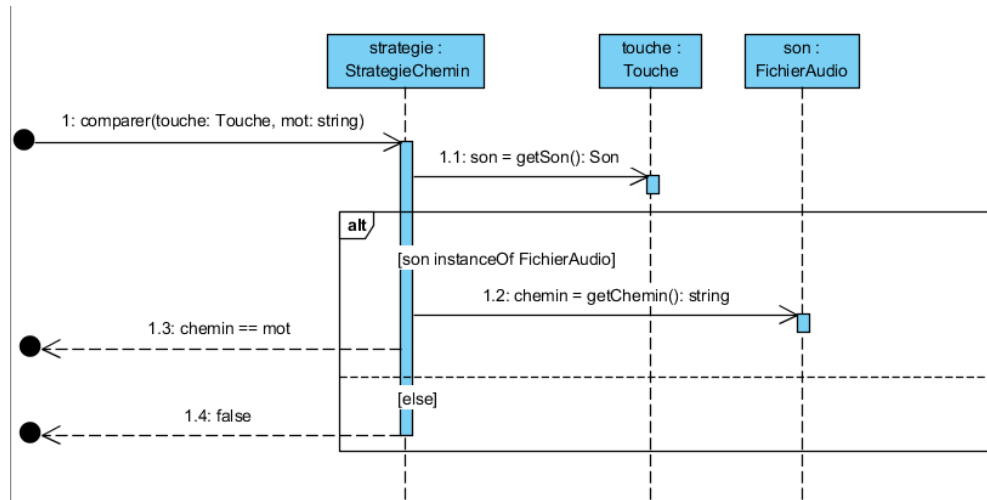
2.1 Rechercher touches



2.1.1 Description

Lorsque le contrôleur reçoit l'appel à `rechercherTouches`, il transfère l'appel à la méthode du même nom dans `Instrument`. Dans l'instrument, on sépare le string de la requête en tableau de strings contenant chacun des mots de la requête. Ensuite, on démarre une boucle qui passe chaque touche de l'instrument. Dans cette boucle, on commence par enlever la surbrillance de la touche au cas où elle serait déjà en surbrillance. Par la suite, dans une double boucle, on compare chacun des mots à chacune des stratégies de recherche (voir 5 prochains diagrammes pour explication de ces stratégies) pour la touche actuelle. Aussitôt qu'on trouve une correspondance, on sort de la double boucle et on met la touche actuelle en surbrillance avant de passer à la prochaine. Si on ne trouve aucune correspondance, on passe à la prochaine sans mettre l'actuelle en surbrillance.

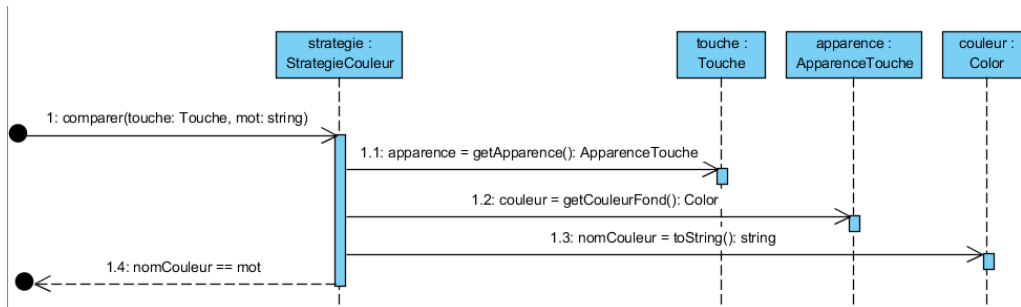
2.2 Stratégie Chemin



2.2.1 Description

On commence par aller chercher l'objet Son dans la touche reçue. Si cet objet n'est pas une instance de la classe FichierAudio, cela signifie qu'il n'a pas le champ chemin. La comparaison retourne donc false. Dans le cas inverse, on va chercher le champ chemin de l'objet (qui représente le chemin du fichier utilisé pour jouer le son), et on le compare au mot reçu. On retourne le résultat de cette comparaison.

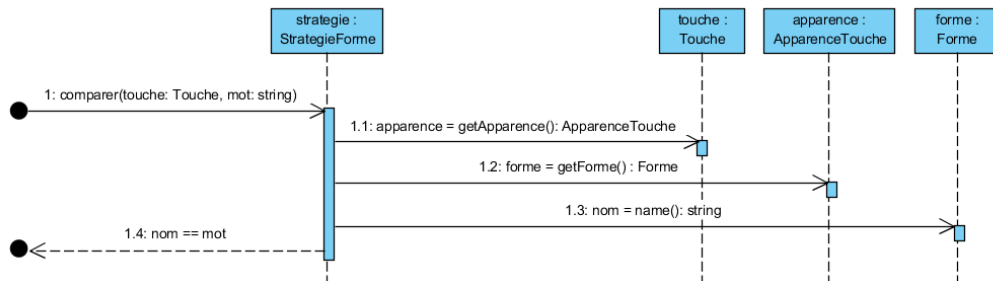
2.3 Stratégie Couleur



2.3.1 Description

On commence par aller chercher l'objet Apparence dans la touche reçue. Dans cette apparence, on va chercher la couleur de fond de la touche. On compare le mot reçu avec cette couleur, convertie en string. On retourne ensuite le résultat de la comparaison.

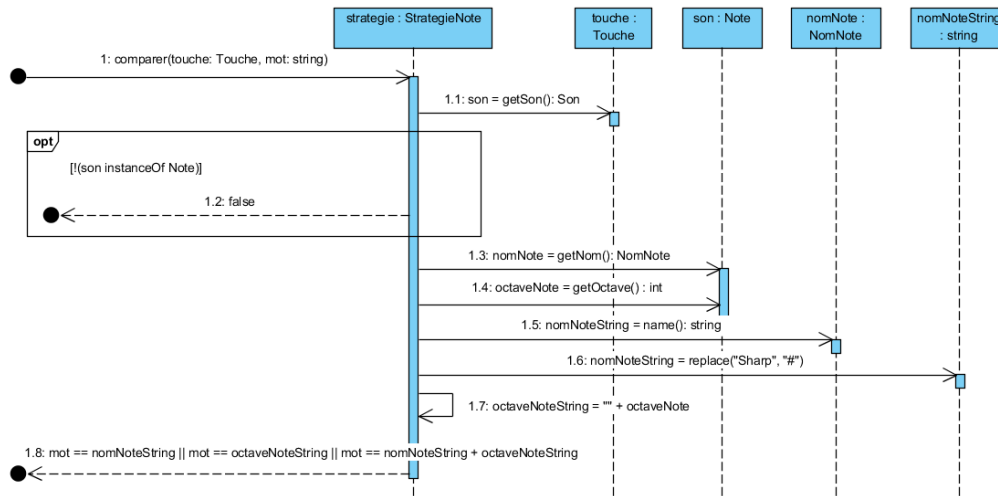
2.4 Stratégie Forme



2.4.1 Description

On commence par aller chercher l'objet `Apparence` dans la touche reçue. Dans cette apparence, on va chercher la forme de la touche (énumération pouvant avoir comme valeur `cercle`, `triangle`, `rectangle`, `pentagone` ou `hexagone`). On compare le mot reçu avec cette forme, convertie en string. On retourne ensuite le résultat de la comparaison.

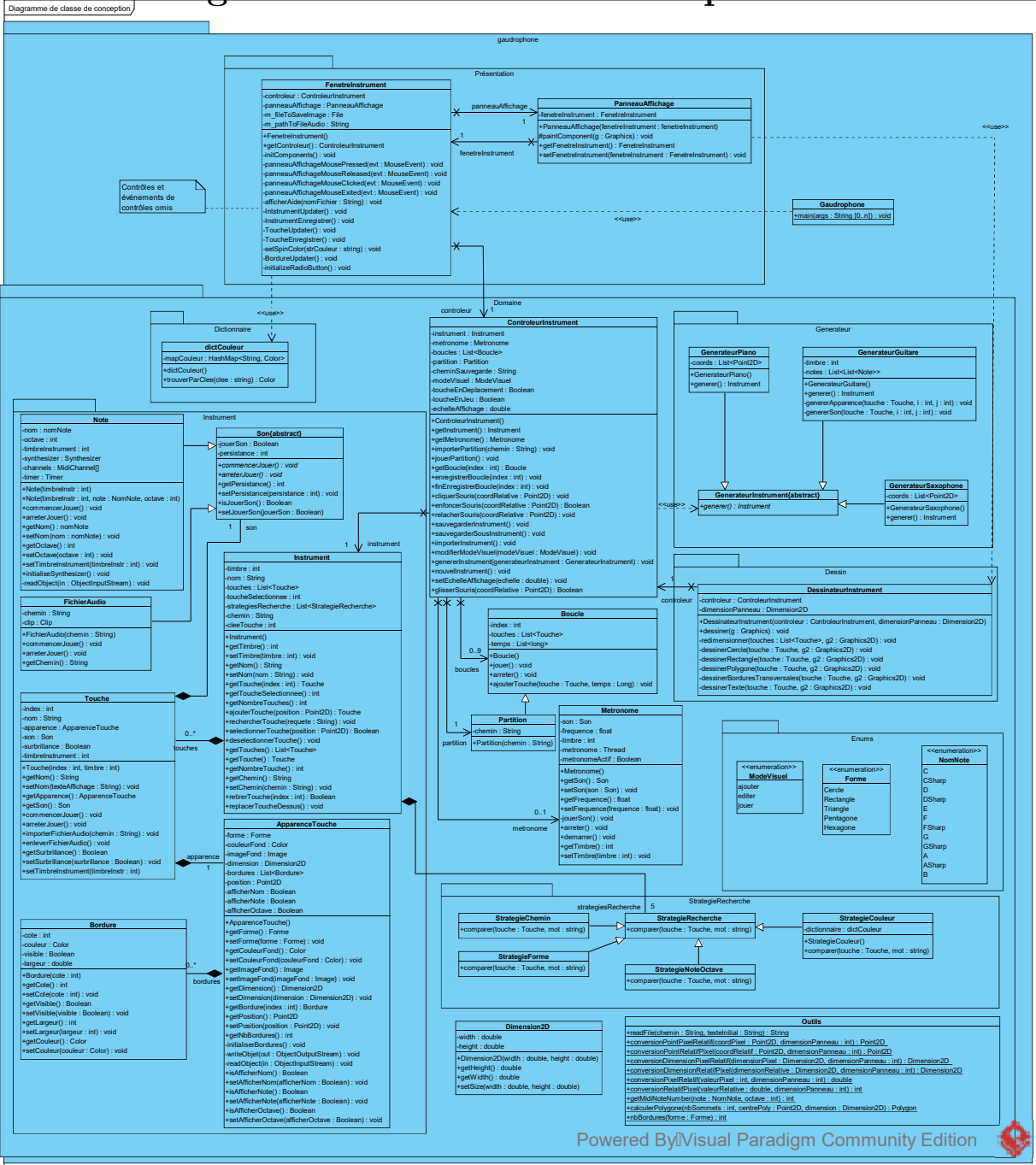
2.5 Stratégie Note et Octave



2.5.1 Description

On commence par aller chercher l'objet `Son` dans la touche reçue. Si cet objet n'est pas une instance de la classe `Note`, la comparaison ne s'applique pas, donc on retourne `false`. Ensuite, on va chercher le nom de la note ainsi que son octave, puis on les convertit tous les deux en `string` pour pouvoir les comparer au mot. Si le mot correspond à la note, à l'octave, ou à la concaténation des deux, on retourne `true`. Sinon, on retourne `false`.

3 Diagramme de classe de conception



4 Contribution des membres

1. Tous, en groupe : Diagramme d'état, diagramme de classes.
2. Samuel Baillargeon : Diagrammes de séquence, générateur guitare, dessinateur instrument, correction de bugs.
3. Émile Gélinas : Interface graphique, correction de bugs.
4. Alexandre Mercier-Aubin : Architecture du logiciel, sérialisation, fonction de recherche, gestion des clics, correction de bugs.
5. Alexandre Venables : Gestion du son, correction de bugs.