

Rapport Projet Informatique Java

Sujet : Tamagotchi

Binôme : MARZEC Adriano et FAVREAU Arthur

Sommaire :

Introduction.....	2
I:Programmation.....	2
II : Scripts.....	3
III : Conclusion.....	7

Introduction :

Nous avons programmé, dans le cadre notre projet informatique, un Tamagotchi. Celui ci devait pouvoir être nourri, soigné, éduqué et il devait être possible de jouer avec lui au BlackJack et au morpion.

Bien sûr, toute extension à ces fonctionnalités était autorisée et même sollicitée, nous avons pour cela plusieurs pistes, notamment l'idée d'un niveau de difficulté qui accélère la diminution des statistiques du Tamagotchi, aussi la possibilité de libérer notre Tamagotchi afin qu'il soit une aide lors des futures parties (cela en fonction de l'intelligence accumulée par celui ci), enfin l'idée d'un pendu comme jeu disponible fût envisagée.

I. Programmation

L'intérêt d'un langage de programmation orienté objet comme Java est que la conception d'un programme peut être morcelée en plusieurs tâches que plusieurs programmeurs peuvent accomplir séparément, c'est à partir de ce principe que nous avons organisé notre travail. Pour ce qui est des rendus de type diagramme et autres supports explicatifs, la répartition du travail fût équilibrée par le temps d'exécution nécessaire, ainsi Arthur s'est occupé du Gantt et du diagramme de cas d'utilisation et Adriano des diagrammes de fréquence et d'activité, le diagramme de classe fût réalisé conjointement par le binôme.

De même, nous avons initialement commencé la programmation sur le même écran, pour les classes primaires : Tamagotchi, Etat, Jeu, Nourriture, Ecole et Medicament. Arthur a ensuite complété ces classes tandis qu'Adriano s'est occupé des classes filles : Viande, Legume, Bonbon, Sirop, Aspirine, Nombre et Mot.

Dans un troisième temps, nous avons passé beaucoup de temps sur les jeux de notre tamagotchi, Arthur s'est occupé du Morpion, de Chatouilles, Balade, Adriano a lui fait le Blackjack. Finalement, nous avons entamé notre main, nous nous sommes relayé, puis des ajustements sur nos premières classes ont eu lieu : Arthur a intégré Ecole au main de Tamagotchi et Adriano a corrigé le Morpion.

C'est alors que nous avons compris que nous avions commis plusieurs erreurs :

- Nous avons conçus nos mini-jeux avec des main propres, ce qui les a rendu impossible à joindre à notre main principal, nous pensions initialement pouvoir lier les package entre eux.
- Notre incapacité a produire une « mémoire tampon », via un fichier sauvegardant les données voulues d'un lancement de programme à l'autre, rend la vie de notre Tamagotchi bien éphémère : une fois le programme arrêté, il disparaît. Nous avons donc, dans un souci ludique, rendu le début de partie plus intéressant en tronquant les statistiques du Tamagotchi nouveau-né.
- Pour la même raison, la tentative d'Adriano de créer une classe Temps, censée faire varier les statistiques du Tamagotchi après un redémarrage du programme ou une action quelconque, fût vaine.

Nous avons finalement privilégié la fonctionnalité aux exigences du cahier des charges, et décidé de rendre trois packages indépendants : Tamago, Morpion et Blackjack fonctionnels indépendamment et tous à usage unique, c'est à dire sans mémoire au-delà d'un lancement.

II :Scripts

Notre package Tamago contient ainsi les classes suivantes :

- Tamagotchi (comprenant le main),
- Etat, qui initialise les variables d'état et fournit les getteurs/setteurs nécessaires à l'obtention des valeurs de santé, satiété, etc...
- Nourriture, qui est un abstract pour les classes filles : Viande, Legume et Bonbon qui initient des variations de l'état du Tamagotchi et qui utilisent les setteurs de Etat pour les appliquer.
- Medicament, qui est un abstract pour les classes filles : Aspirine et Sirop
- Jeu, qui est un abstract pour les classes filles : Balade, Chatouilles, Morpion et Blackjack.
- Temps, qui n'est pas fonctionnelle et qui devait intervenir dans toutes les autres classes, vu que le temps est censé se dérouler à chaque instant.

Les classes du package Morpion :

- Symbole, qui sert à instancier un symbole lié à un joueur
- Joueur, qui lie un joueur à son symbole
- CaseGrille, qui crée une case de la grille de morpion en l'identifiant, (on utilise en réalité une liste plus tard), ses méthodes permettent aussi de placer un symbole et de connaître son propriétaire, vide() permet de vider les cases.
- Grille, qui initialise la liste de type CaseGrille qui constitue la grille, permet de vérifier si une ligne/colonne/diagonale de trois symboles est formée, complet() permet aussi de vérifier si la grille est pleine et d'automatiquement la vider si c'est le cas.
- Entree, qui est une classe auxiliaire permettant aux entrées du joueur d'être valide dans le contexte du Morpion
- Jeu, qui utilise toutes les fonctions vues précédemment pour effectuer des tours de jeu avec 2 joueurs.
- Morpion, qui contient un main et hérite de Jeu.

Les classes du package BlackJack :

- Valeur, qui est une énumération des valeurs qu'une carte peut prendre
- Figure, qui énumère les figures ou « couleurs » qu'une carte peut prendre
- Carte, qui utilise Valeur et Figure afin de créer une carte aléatoire
- Deck, qui utilise Carte afin de créer un deck de 52 cartes, cette classe se compose de multiples méthodes telles que piocher() ou mélanger() afin d'interagir comme c'est nécessaire avec les decks qui seront nos pioches ou nos mains
- Blackjack qui comporte le main et instancie trois decks : les mains de la banque et du joueur, et la pioche, qui vont interagir ensemble durant le main.

Notre Tamagotchi fonctionne sous une forme interactive avec le joueur, le programme demande continuellement au joueur ce qu'il veut faire et lui propose les choix qui sont disponibles . Une multitudes de String racontent les conséquences des actes du joueur au cours de la partie.

De par les changements opérés le diagramme de classes rendu précédemment est devenu obsolète. Nous avonx donc maintenant 3 diagrammes de classes : 1 pour chaque jeu.

Diagramme de classes du Tamagotchi :

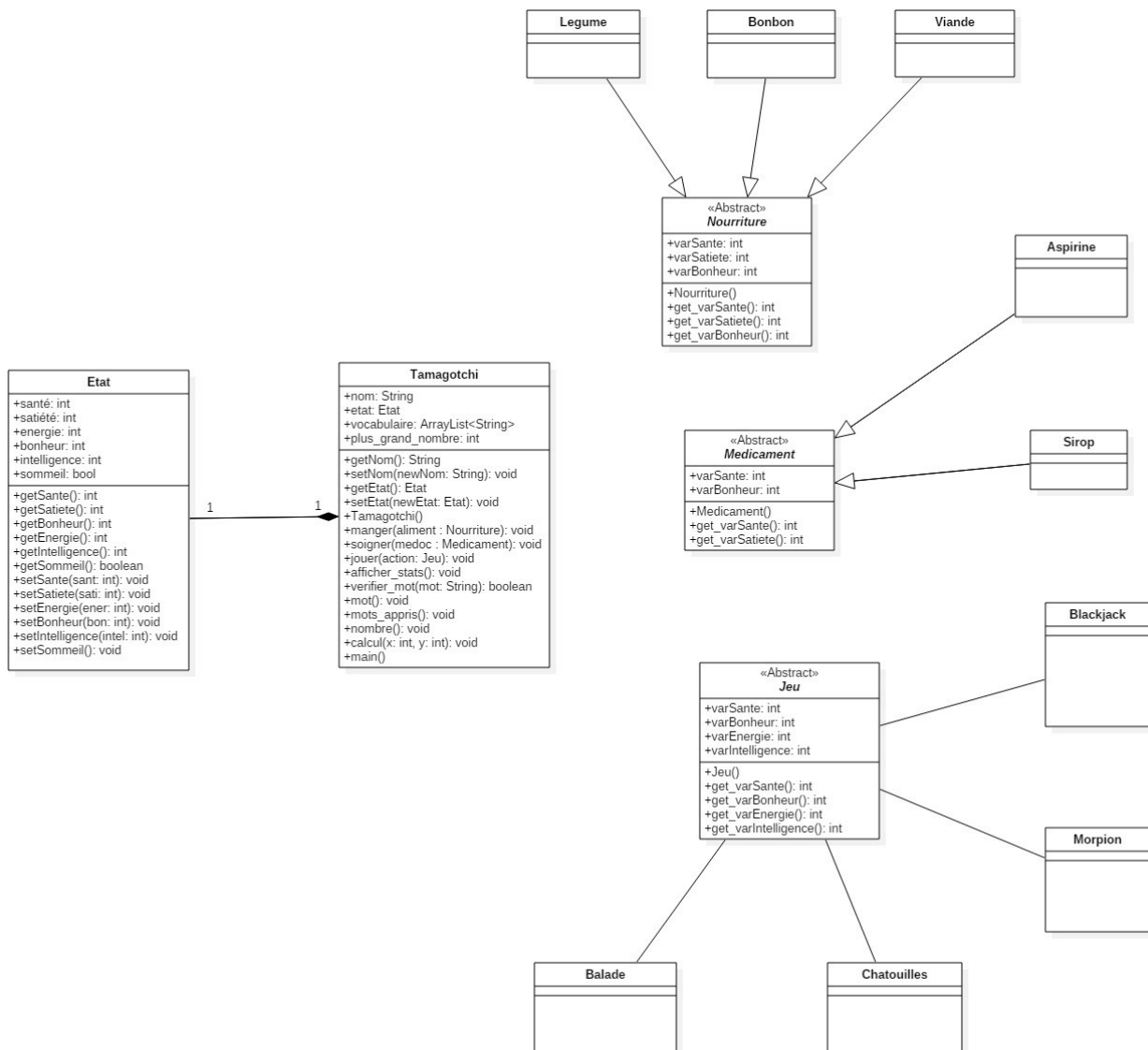


Diagramme de classes du Morpion :

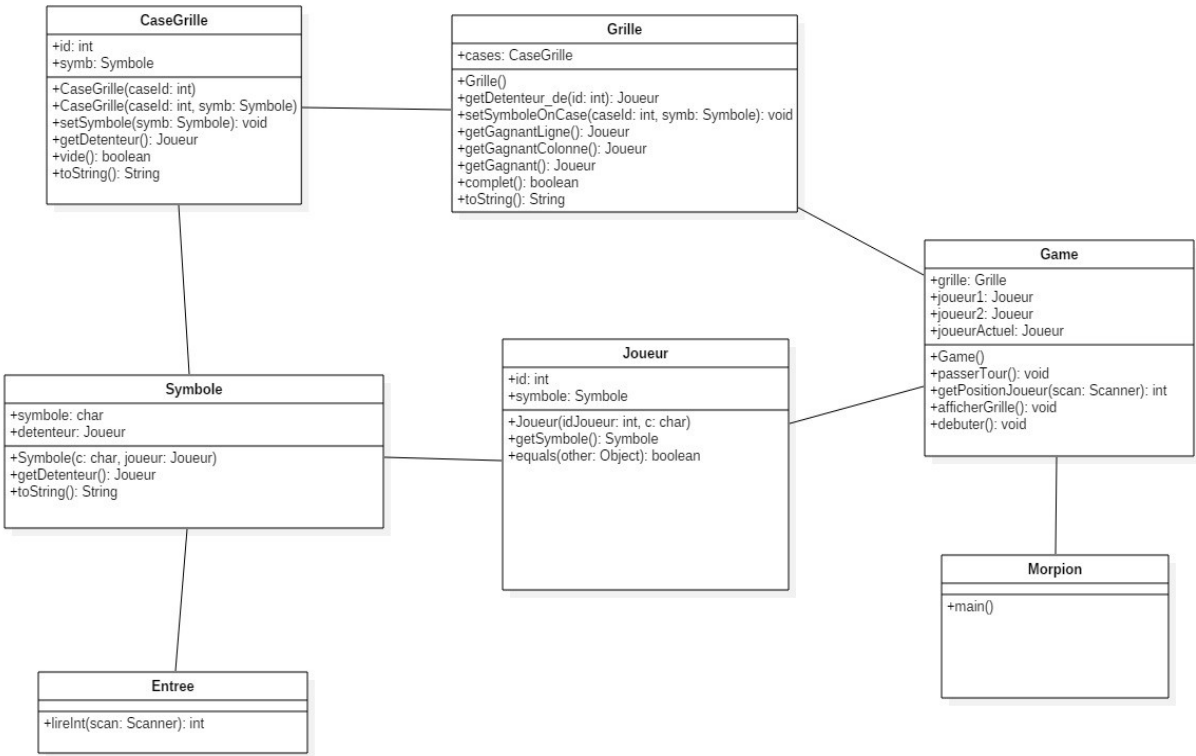
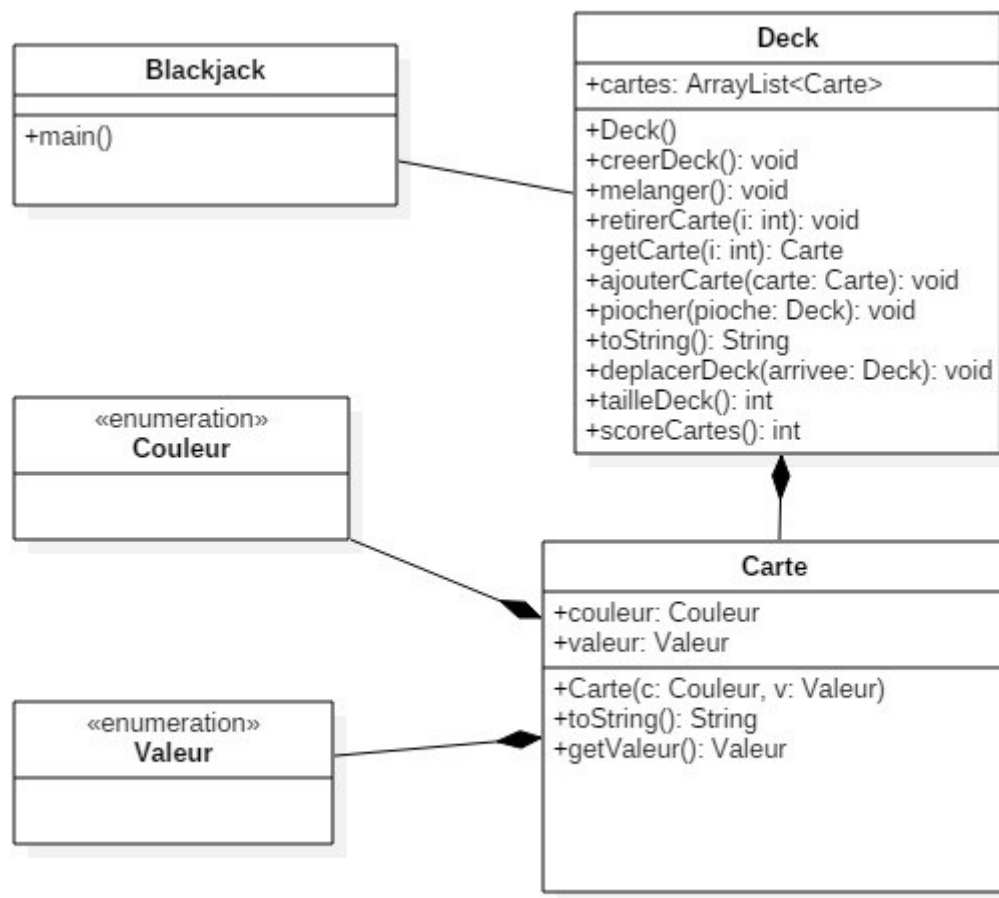


Diagramme de classes du Blackjack :



III :Conclusion

Nous pensions pouvoir être ambitieux dans les fonctionnalités à apporter à notre Tamagotchi, nous avons probablement sous-estimé ce travail puisque nous ne nous attendions pas à ce que les fonctions du cahier des charges soient en soi une difficulté insurmontable dans le délai qui nous était imparti. La pratique du java nous a certainement fait progresser mais pas suffisamment vite pour accomplir ce projet pourtant noté comme étant le plus facile. Peut être qu'aujourd'hui serions nous capable de mener un tel projet à bien avec les mêmes conditions.