

Projet de Conception agile de projets informatiques et génie logiciel

CLONE PROJET : git clone https://github.com/Manu-RMT/TP_M1_JAVA.git

Table des matières

1 - Présentation du projet	2
2 - Structure du projet.....	2
2.1 Modèle Modèle-Vue-Contrôleur.....	2
2.2 Singleton.....	4
2.3 Fonctions implémentées	4
Conclusion	5

1 - Présentation du projet

Le projet consiste à reprendre un code rédigé au préalable, à restructurer et à développer en binôme. Le code sert à lancer un jeu de balle au prisonnier. Le développement du projet a pu être réalisé sur Eclipse et sur IntelliJ.

Initialement, le code était constitué de plusieurs classes. Il permettait d'afficher une fenêtre avec un joueur en haut et un autre en bas de celle-ci. Les joueurs étaient amovibles de gauche à droite et on pouvait également changer la direction des flèches noires qui étaient associées aux joueurs.

Nous avons commencé par modifier les touches de contrôle d'un des deux joueurs. Ensuite nous avons dissocié les 2 joueurs, en considérant le joueur du haut comme une IA et en gardant le contrôle sur celui du bas. Une fois avoir créé deux classes différentes pour les joueurs, nous avons pu ajouter des joueurs en créant de nouveaux objets à partir des classes. On est parti d'une classe commune regroupant les caractéristiques communes à tous les joueurs (Player). De cette classe, deux autres sont hérités : IA et Human qui ont des caractéristiques propres à leur sous-classe. Nous avons gardé deux joueurs que l'utilisateur peut contrôler. Nous avons réussi à afficher une balle que nous n'avons toutefois pas réussi à faire lancer par les joueurs. Nous avons voulu développer l'IA et on leur à développé des déplacements aléatoires.

Nous avons pu collaborer à deux grâce à GIT qui nous a permis de travailler à distance et d'apporter nos modifications en envoyant nos fichiers par le GIT push. Cela a permis de voir les avancements de chacun, corriger en cas d'erreur grâce au pull request avant de faire le merge avec le projet distant.

2 - Structure du projet

2.1 Modèle Modèle-Vue-Contrôleur

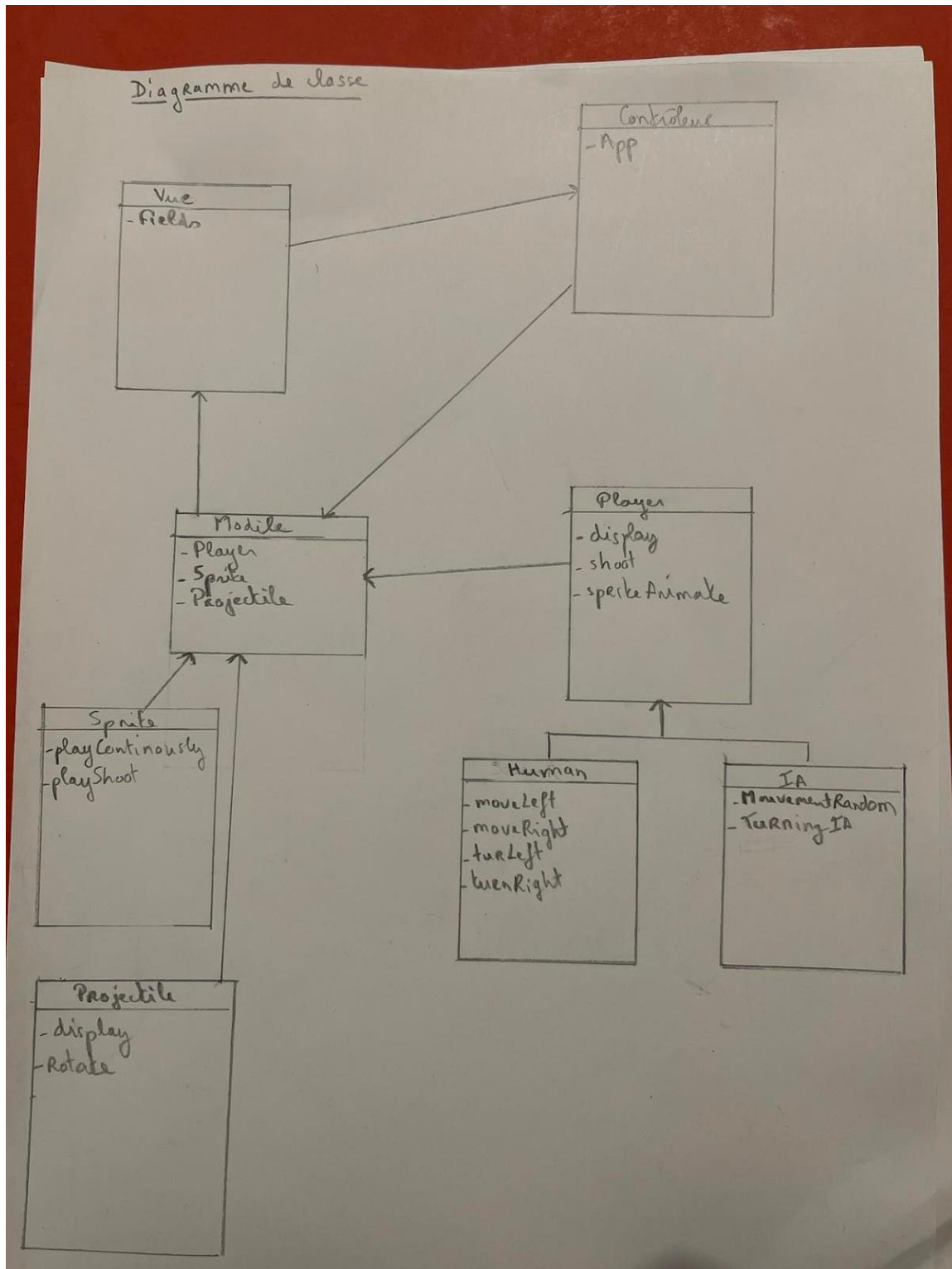
Dans ce projet, on a voulu utiliser un modèle pattern qui est très utilisé lors d'un projet qui est le modèle Pattern MVC.

Cette structure permet de catégoriser son code en séparant les modèles, les vues et les contrôleurs.

Grâce à ce modèle pattern, on pourra par la suite :

- avoir une meilleure visibilité sur son code
- délayer plus facilement le projet à un autre collègue développeur
- réduire le coût de réalisation d'extension du projet
- maintenir plus facilement son code

Voici ci-dessous une illustration qui résume l'architecture de notre projet :



Modèle :

Les modèles correspondent aux différents objets dont on a besoin pour jouer.

Ici, on a :

- Player
- IA
- Human
- Projectile

Vue :

La vue va nous permettre d'afficher les éléments à l'écran.

Pour cela, on a la classe Fields qui permet de rafraîchir l'affichage du jeu toutes les 6 secondes environ ainsi que de s'occuper de l'animation de différents éléments

Contrôleur :

Concernant le contrôleur, son rôle consiste à faire le lien entre les modèles et la vue.

La classe App fait la correspondance entre les classes du modèle et celles de la vue.

2.2 Singleton

Nous avons ensuite voulu appliquer un second patron de conception à notre projet. On voulait restreindre l'instanciation de la classe de la balle à un seul objet.

On a modifié la classe Projectile afin qu'elle ne puisse s'instancier qu'à une seule reprise afin de s'assurer de l'unicité de la balle.

2.3 Fonctions implémentées

Dans IA :

- MovementRandom : permet de mouvement aléatoire de l'IA, soit vers la gauche, soit vers la droite
- Turning_IA : gère l'angle de tire du joueur vers la gauche et vers la droite

Dans Human :

- MoveRigth : bouge le joueur vers la droite
- MoveLeft : bouge le joueur vers la gauche
- TurnRigth : bouge l'angle du joueur vers la gauche
- TurnLeft : bouge l'angle du joueur vers la droite

Dans Projectile :

- Deplacement : gère le déplacement de la balle
- display : permet d'afficher la balle
- rotate : gère la rotation de la balle

Conclusion

Ce projet nous a permis de nous entraîner à la conduite de projet ainsi que sur la collaboration du développement du code. Nous avons dû nous organiser afin d'avancer le plus efficacement possible. C'était une première expérience sur Java, nous avons donc dû observer le fonctionnement de ce nouveau langage. On a eu du mal avec le langage du programme, ce qui nous a fait perdre beaucoup de temps dans la réalisation.

Nous avons commencé par observer le code qui nous a été fourni et à partir de celui-ci nous nous sommes demandés quelles étaient les étapes à franchir avant d'arriver jusqu'aux objectifs que nous nous étions fixés. Une fois que ceci était clair nous sommes passés au développement du code.