POLYTECH SORBONNE

SORBONNE UNIVERSITÉ

COMPTE RENDU PROJET C++

VICTOR ZHENG
CHRISTOPHER AL KHAWAND

Partie 1 : Description de l'application

● Cartes du jeu

Nom du joueur : Lebron James
Poste du joueur : [SF] (Small Forward)
ATQ (Valeur d'attaque) : 97
DEF (Valeur de défense) : 97
VIT (Valeur de vitesse) : 96

● Comment jouer

Appuyez sur Start pour commencer une partie.
Une partie se joue en 12 manches.

Au début de chaque manche, votre adversaire choisit au hasard un joueur de son équipe.
C'est à vous ensuite de choisir un joueur de votre équipe qui devra affronter celui de votre adversaire.
De plus, vous devez choisir si vous voulez utiliser la valeur d'ATQ, de DEF ou de VIT du joueur que vous avez choisit. Appuyer ensuite sur Play pour valider pour action.
Si vous avez choisit d'utiliser la valeur d'ATQ de votre joueur alors votre adversaire utilisera la valeur de DEF de son joueur.
Si vous avez choisit d'utiliser la valeur de DEF de votre joueur alors votre adversaire utilisera la valeur d'ATQ de son joueur.

Victor ZHENG
Christopher AL KHAWAND

Si vous avez choisit d'utiliser la valeur de VIT de votre joueur alors votre adversaire utilisera la valeur de VIT de son joueur.

EXEMPLE : Si vous utilisez la valeur d'ATQ de votre joueur,

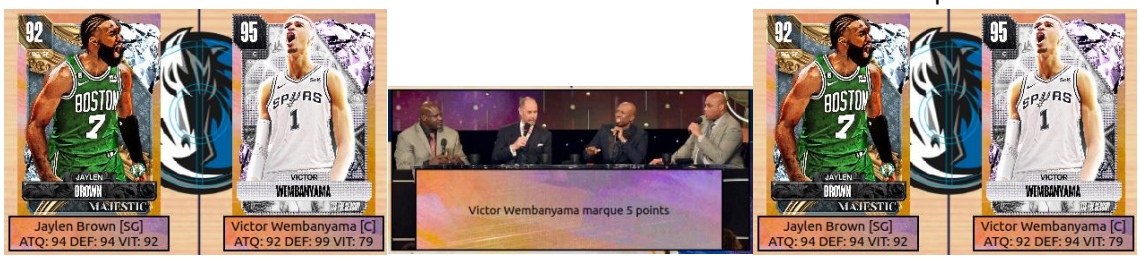
Victoire : Si la valeur d'ATQ de votre joueur est supérieure à la valeur de DEF du joueur de votre adversaire, vous remportez cette manche et vous marquez un nombre de points égals à la différence de ces deux valeurs. Votre joueur perd aussi cette différence en ATQ, DEF et VIT.

Avant

Après

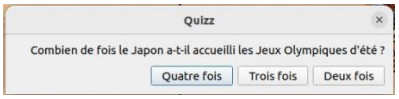
Défaite : Si la valeur de DEF de votre joueur est supérieure à la valeur d'ATQ du joueur de votre adversaire, votre adversaire remporte cette manche et il marque un nombre de points égals à la différence de ces deux valeurs. Le joueur de l'adversaire perd aussi cette différence en DEF.

Avant



Après

Egalité : Si la valeur d'ATQ de votre joueur est égale à la valeur de DEF du joueur de votre adversaire, alors vous allez devoir répondre à une question portée sur les Jeux Olympiques avec à chaque fois trois réponses possibles.



Si vous répondez juste alors vous marquez 3 points et votre joueur perd 3 en ATQ, DEF et VIT.
Si vous répondez faux, alors vous marquez 0 point et votre joueur perd 10 en ATQ, DEF et VIT.

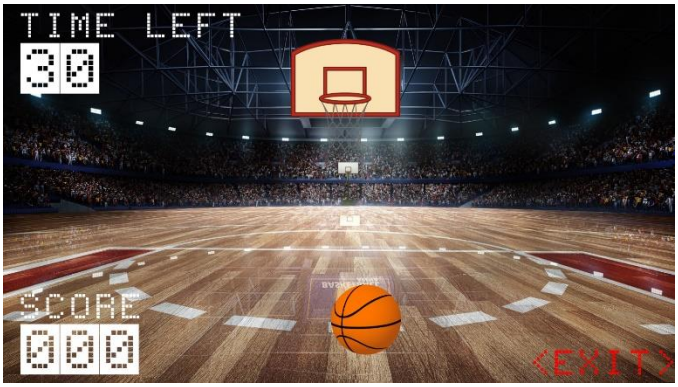
Le poste du joueur est important car celui-ci lui permet d'obtenir une augmentation de +2 en ATQ, DEF et VIT face à un joueur du poste opposé.
[PG] < [SG] < [SF] < [PF] < [C] < [PG]

Victor ZHENG
Christopher AL KHAWAND

Coach Boost : Vous pouvez utiliser cette capacité à n'importe quel instant de la partie.
Elle vous permet de connaître les valeurs d'ATQ, de DEF et de VIT des joueurs de votre adversaire.
Vous avez aussi la possibilité de choisir le nombre de joueur dont vous souhaiter révéler les valeurs.
En contrepartie, utiliser cette capacité permet aussi à votre adversaire de marquer des points proportionnels au nombre de joueur choisit, et donc lui permettra de prendre de l'avance sur la partie. Utilisez le donc à bon escient.

Vous pouvez très bien gagner la partie sans utiliser cette capacité si vous pensez avoir une bonne connaissance des valeurs d'ATQ, de DEF et de VIT des joueurs de votre adversaire.

Extra Stage : A la fin des 12 manches, une autre fenêtre de jeu se lance. Vous avez 30 secondes pour mettre le plus de points additionnels possibles.



Ainsi, celui qui aura marqué le plus de point entre vous et votre adversaire remporte la partie.

Partie 2 : Mettre en valeur l'utilisation des contraintes

- Au moins 8 classes : Il y a plus de 8 classes dans notre application (cf. Diagramme UML)

- 3 Niveaux de hiérarchie : Notre application contient 3 niveaux de hiérarchie (cf Diagramme UML)

- 2 Fonctions virtuelles différentes et utilisées à bon escient : Chaque instance de la classe « Joueur » possède ces trois méthodes : atq, def et vit. Il s'agit de fonctions virtuelles car les classes qui héritent de la classe « Joueur », ont un fonctionnement différent pour les méthodes atq, def et vit.

- 2 Surcharges d'opérateurs : Nous avons implémenter trois surcharges d'opérateurs dans la classe « Joueur »,
  - « << » : qui permet d'afficher le nom, prénom et le nombre de points marqués de celui-ci
  - « ++ » : qui permet d'incrémenter le nombre de points marqués par un Joueur.
  -« += » : qui permet d'augmenter le nombre de points marqués par un Joueur par un nombre spécifique de points.

Victor ZHENG
Christopher AL KHAWAND

● 2 Conteneurs différents de la STL : Nous avons utilisé deux conteneurs de la STL, des « vector » et des « map ».

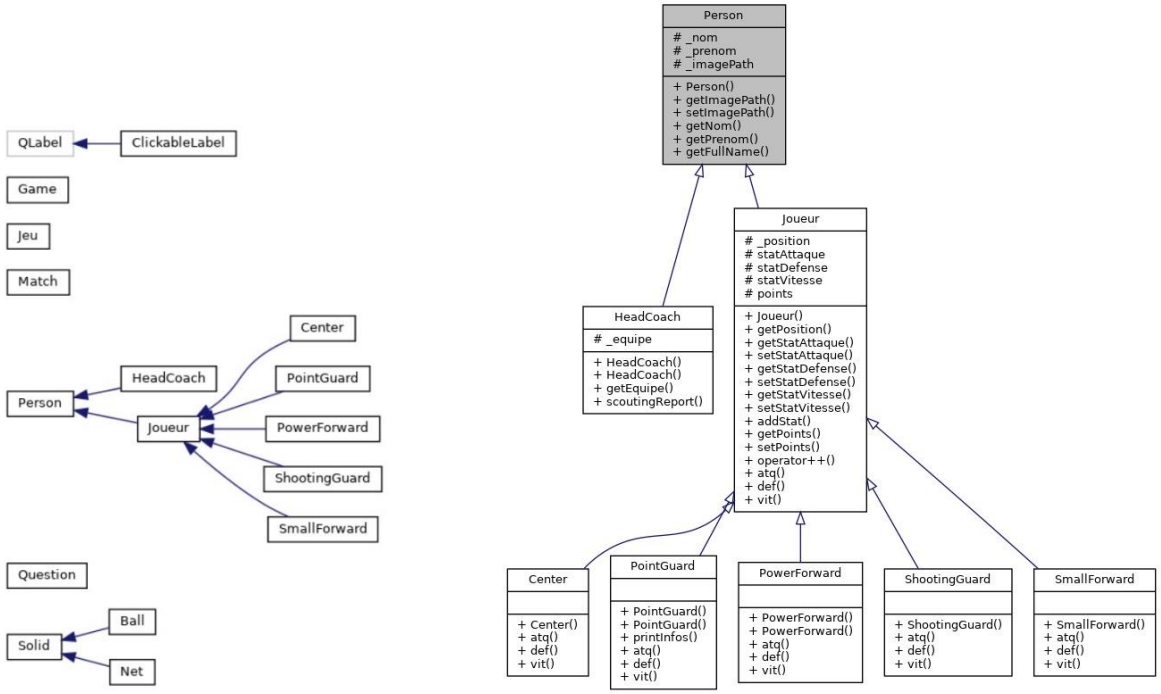
● Pas de méthodes/fonctions de plus de 30 lignes : Nous avons bien fragmenté nos fonctions. Ainsi, aucune d'entre elles ne fait plus de 30 lignes.

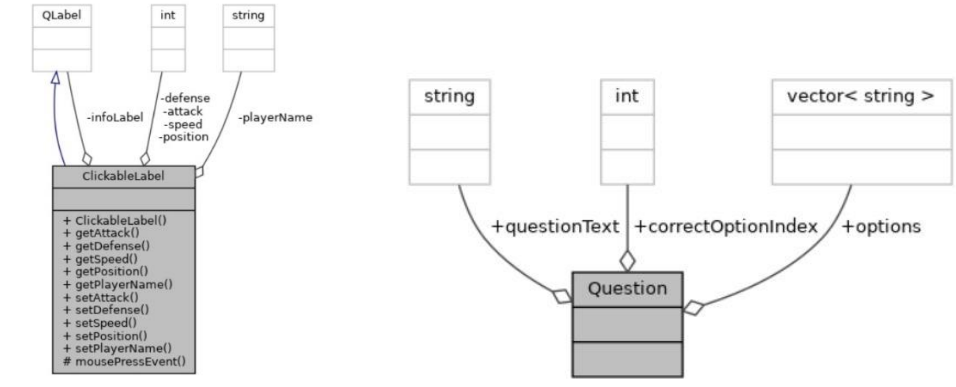
● Utilisation d'un CMakeLists : Nous avons utilisé un CMakeLists.

● Utilisation de tests unitaires : Pour tester le bon fonctionnement de nos méthodes dans nos classes, nous avons écrit des tests pour vérifier si nous obtenons bien les résultats attendus.

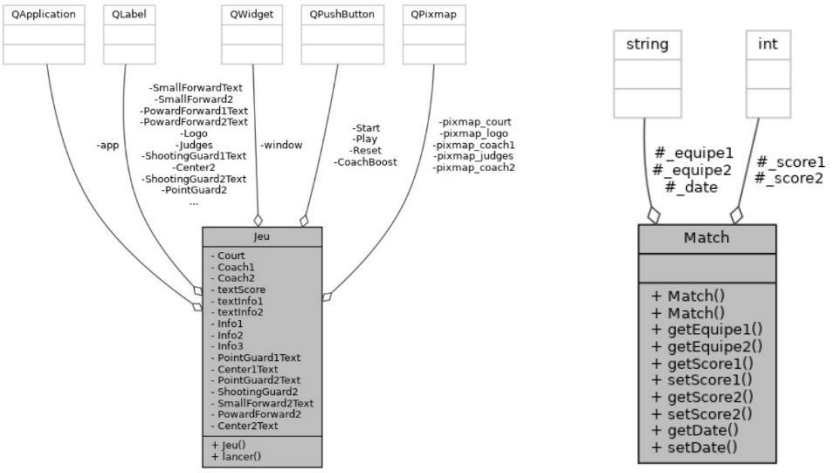
Partie 3 : Diagramme UML de l'application

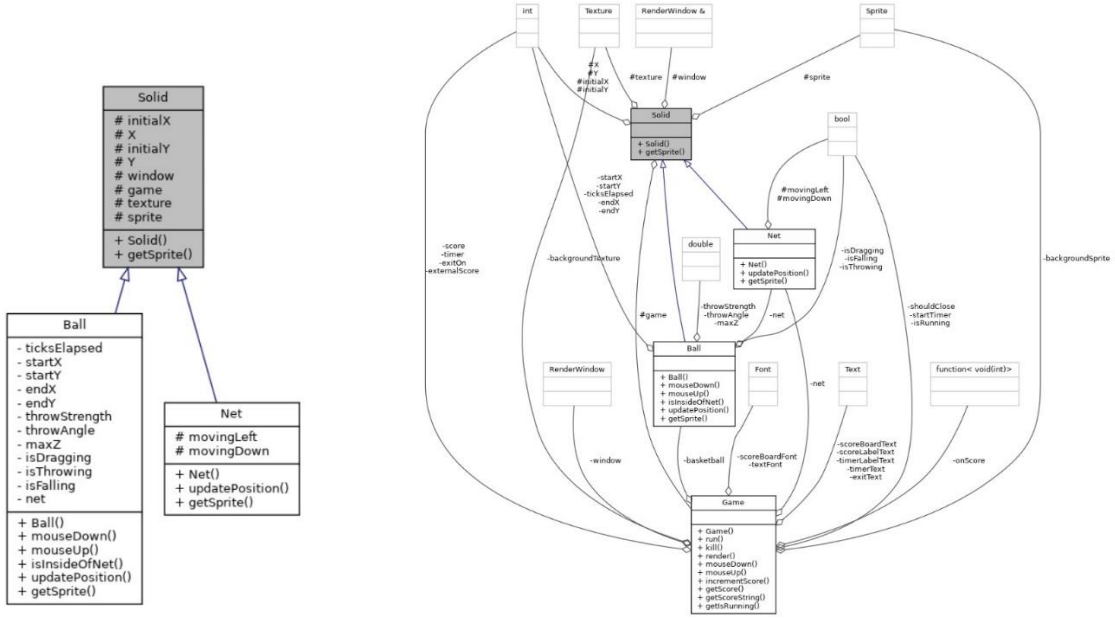
Les diagrammes UML suivants ont été faits avec Doxygen :





Victor ZHENG
Christopher AL KHAWAND





Partie 4 : Procédure d'installation et d'éxécution du code

Pour éxécuter le code, il faut installer la librairie Qt et SFML avec ces lignes de commande :

```
sudo apt-get install qt5-default
sudo apt-get install libsfml-dev
```

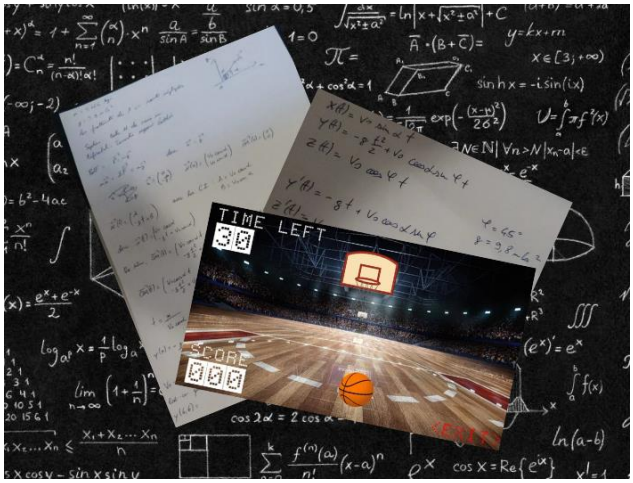
Puis allez dans le directoire du projet, ouvrez le terminal et exécuter la commande « ./run.sh » pour compiler le code et lancer le jeu.
Lancer « ./test.sh » pour compiler les tests et lancer doctest.

Partie 5 : Les parties de l'implémentation dont vous êtes les plus fiéres

Nous sommes particulièrement fières de notre interface graphique qui a été implémentée avec Qt mais aussi de la jouabilité de notre jeu. En effet, notre application s'assimile à un casse-tête où l'on doit trouver un équilibre entre marquer des points et gérer les statistiques de nos joueurs. Car plus

on marque de points plus nos joueurs deviennent faibles pour les prochaines manches. Ainsi, une bonne connaissance du jeu et des statistiques des joueurs vont vous permettre de gagner plus facilement.

De plus, notre « Extra Stage » ,qui consiste à lancer la balle de basket dans le panier à l'aide de la souris, est une autre partie de notre implémentation dont nous sommes fières. Cette partie nous a demandé de nous replonger dans nos anciens cours de physique pour connaître l'équation de la trajectoire de la balle pour simuler le lancer franc.



Victor ZHENG
Christopher AL KHAWAND