.Net CardGame

Contexte du projet :

Ce projet est à réaliser dans le cadre du module .Net afin de faire nous faire découvrir le .net autour d'un jeu de cartes, nous avons choisi de recoder une bataille.

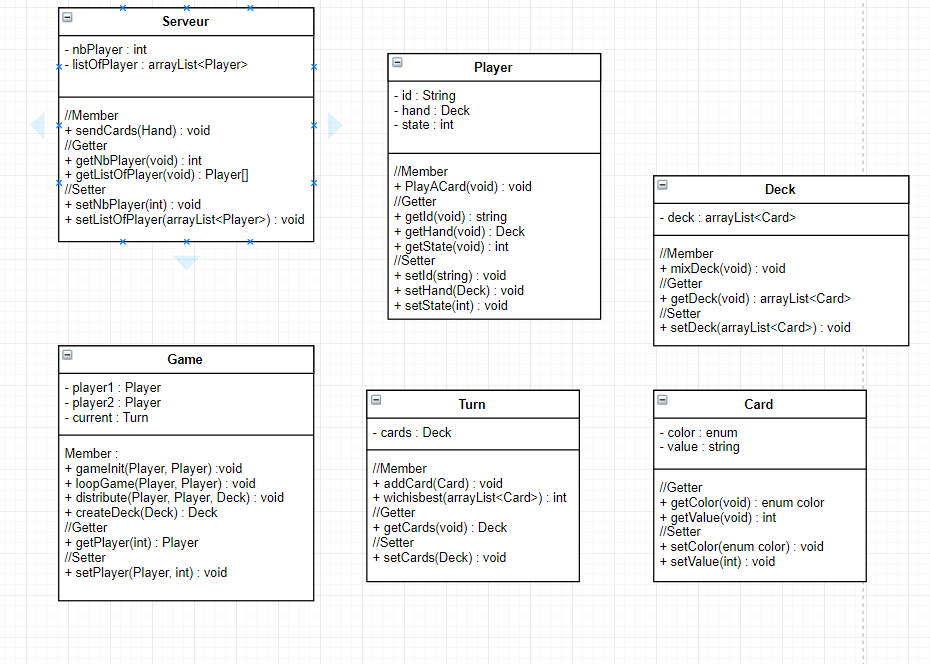
La bataille est un jeu de cartes se jouant à deux joueurs. En l'occurrence, il s'agira ici d'une bataille fermée, c’est-à-dire que la totalité des cartes est partagée entre les deux joueurs, ceux-ci n'ayant qu'à jouer la carte sur le dessus de leur paquet par la suite. Les joueurs ne peuvent en aucun cas connaître leurs cartes ou décider laquelle jouer.

Architecture :

Dans ce projet, nous distinguons plusieurs parties :

* Un serveur : met en place la communication avec les clients futurs, détient les informations sur le jeu, les cartes des joueurs, etc..
* Des clients : représentant les joueurs.

Diagramme de classes :



Diagrammes de séquences :

Cas d'utilisation :

Dépendances, librairies :

Dans le cadre de ce projet, nous avons développer ce programme sur Visual studio community 2017, avec le framework Resharper 2017 2.2.

Pour gérer toutes les connections : Network ComsDOTNET

Pour les tests unitaires, nous avons utilisé Xunit. Pour ajouter Xunit sur Visual studio community :   
-Cliquez droit sur référence, "Manage NuGet Packages..." , puis tapez dans la recherche "xUnit" et sélectionnez :  
-"xunit.runner.console" version 2.3.1  
-"xunit" version 2.3.1  
-"xunit.analyzers" version 0.7.0

Comment lancer le programme ?

RFC du serveur et des clients :

*Server :*

-server -> client: "000: YOU ARE CONNECTED!"

-server -> client: "001: Waiting for other players to connect."

-server -> client: "002" Envoyer un message au client

-server -> client: "003" Demande a client de jouer une carte

-server -> client: "004: Sending your wins." Envoie des cartes gagnées

-server -> client: "009: YOU LOOSE THE GAME."

-server -> client: "010: YOU WIN THE GAME."

-server -> client: "011: SCORE [NBR] - [NBR]."

-server -> client: "666: FATAL ERROR."

*Client :*

-client -> server: "100: Message received."

-client -> server: "101: Deck received."

-client -> server: "103: Sending card."