Roland DENIZOT
Zachary CHENOT

Final Project – A Monopoly™ Game – Design Pattern and Software Development Process

1. Introduction

This project is a simplified Monopoly Game coded in C# using design patterns. This one differs from a classic Monopoly because we focused on the jail and the conditions to go and to go out of the jail. This project is part of the end of semester project of the course named "Design patterns and software development process". Our group is made up of Zachary CHENOT and Roland DENIZOT, students in fourth year at ESILV in group DIA2.

2. Design Hypotheses

Singleton :
The purpose of using a singleton is to be sure there is only one instance of the board existing.

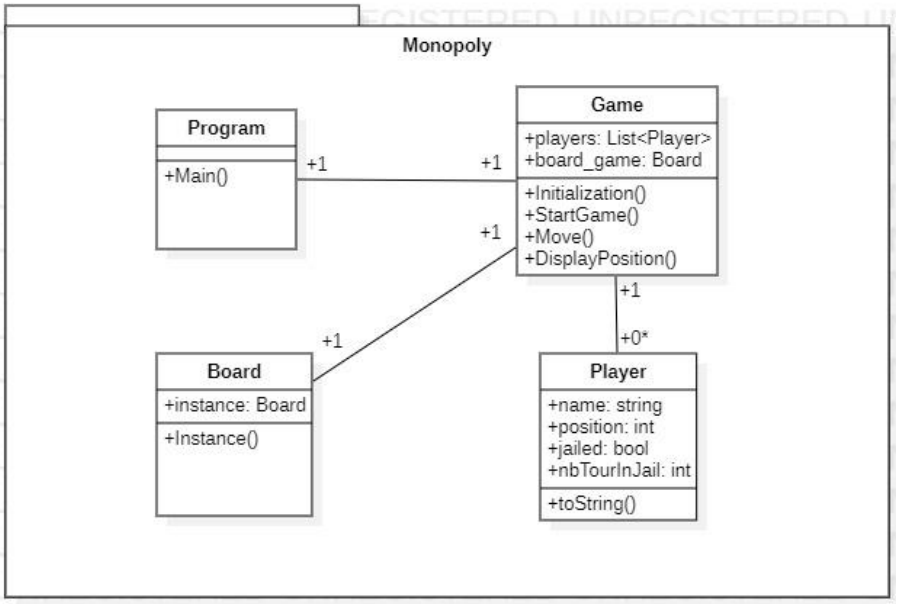
```
5 références
public class Board
{

        private static Board instance = null;

        0 références
        public static Board Instance
        {
            get
            {
                if (instance == null)
                {
                    instance = new Board();
                }
                return instance;
            }
        }
}
```

End of game :
Since the game appears to be endless, we propose the player to end the game after each turn.

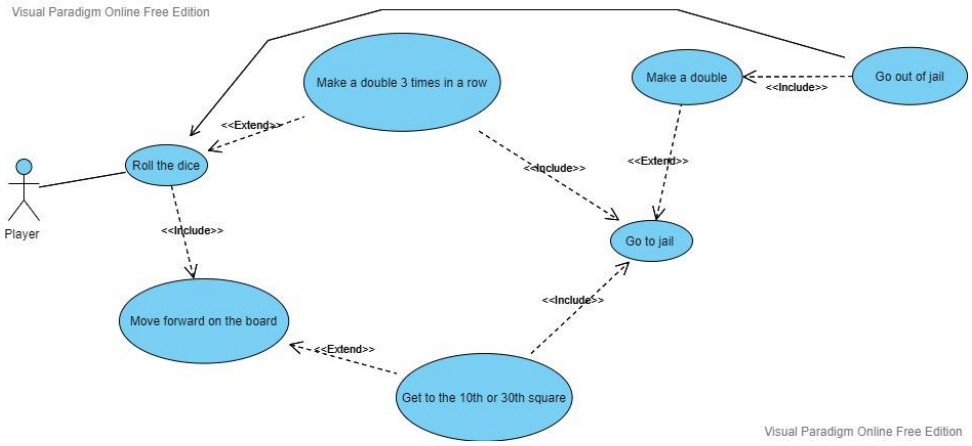
DisplayPosition(Player player):
This function isn't really necessary but it's definitely better for the players to know where they are on the board.

3. UML Diagrams

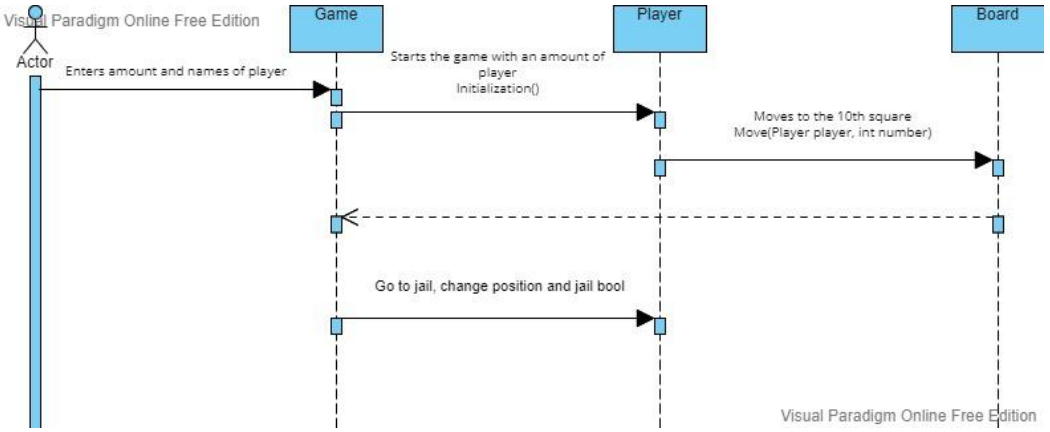
- Class Diagram of the solution:



- Use Case Diagram:



- Sequence Diagram:



4. Test Cases

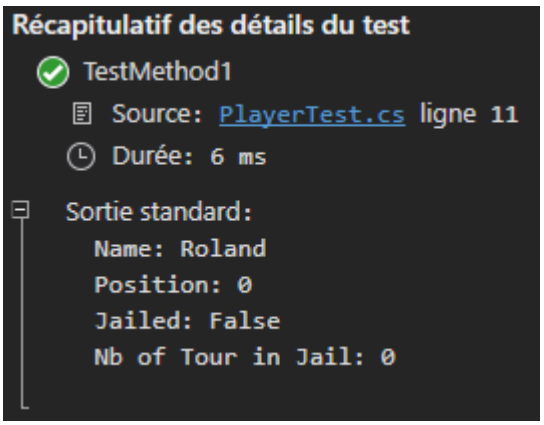
In this project, we executed two unit tests:

- The first test is creating a player named ‘Roland’ and we tested the function toString() of the class Player. As the default values of a player are 0 for position, False for jailed and 0 for number of tour in jail, the expected output is:
  ‘Name: Roland
  Position: 0
  Jailed: False
  Nb of Tour in Jail: 0’

  Code:

```
[TestMethod]
0 références
public void TestMethod1()
{
    Player player1 = new Player("Roland");
    player1.toString();
}
```

  Output:

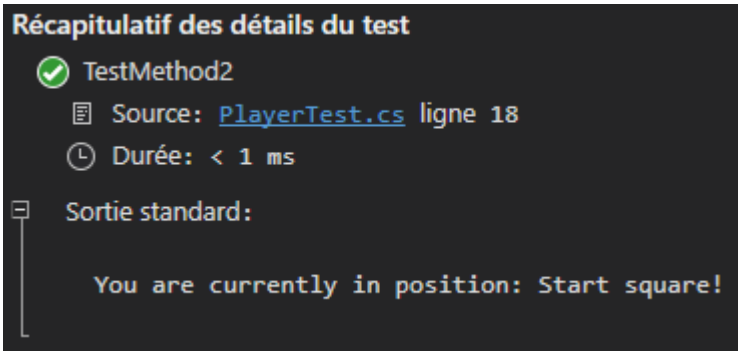


- The second test is initializing a new game named ‘monop’ and a player named ‘Roland’. We tested the function DisplayPosition() of the class Game. As the default position of a player is 0, corresponding to the start square, the expected output is : ‘You are currently in position: Start square!’

  Code:

```
[TestMethod]
0 références
public void TestMethod2()
{
    Game monop = new Game();
    Player player1 = new Player("Roland");
    monop.DisplayPosition(player1);
}
```

Output:


Récapitulatif des détails du test
TestMethod2
Source: PlayerTest.cs ligne 18
Durée: < 1 ms
Sortie standard:
You are currently in position: Start square!