Tien Nguyen

PROG 101 FA22

Midterm Project

Adventure Game: The World of Mystic Creatures

In *The World of Mystic Creatures,* you play an anonymous protagonist who accidentally fell off a cruise and got lost in the under-ocean world where there's people and sea animals live harmoniously. You need to find Turlie The Sorcerer*,* who lives in the SeaWeed Forest. He has the power to take you back home. The journey to find him is challenging, however, you will meet with friendly villagers of the Oceano Village on the way there. The story begins with you wandering around under the deep blue sea, and suddenly you found a sign beside the road.

On the way to find the sorcerer, there will be instructions and clues, which are given by Miss Octopy, the old flounder lady, the little mermaid, and Mr. SeaHorse. The player has to read the instructions carefully, they will help with the decision-making. The player will receive a pouch while playing the game. The pouch stores items/weapons to defeat the monsters in the final round. The player can only use the weapon once, but they can retry the fight until they get more than 10 points. If they win the fight, they will be able to meet Turlie The Sorcerer and win the game.

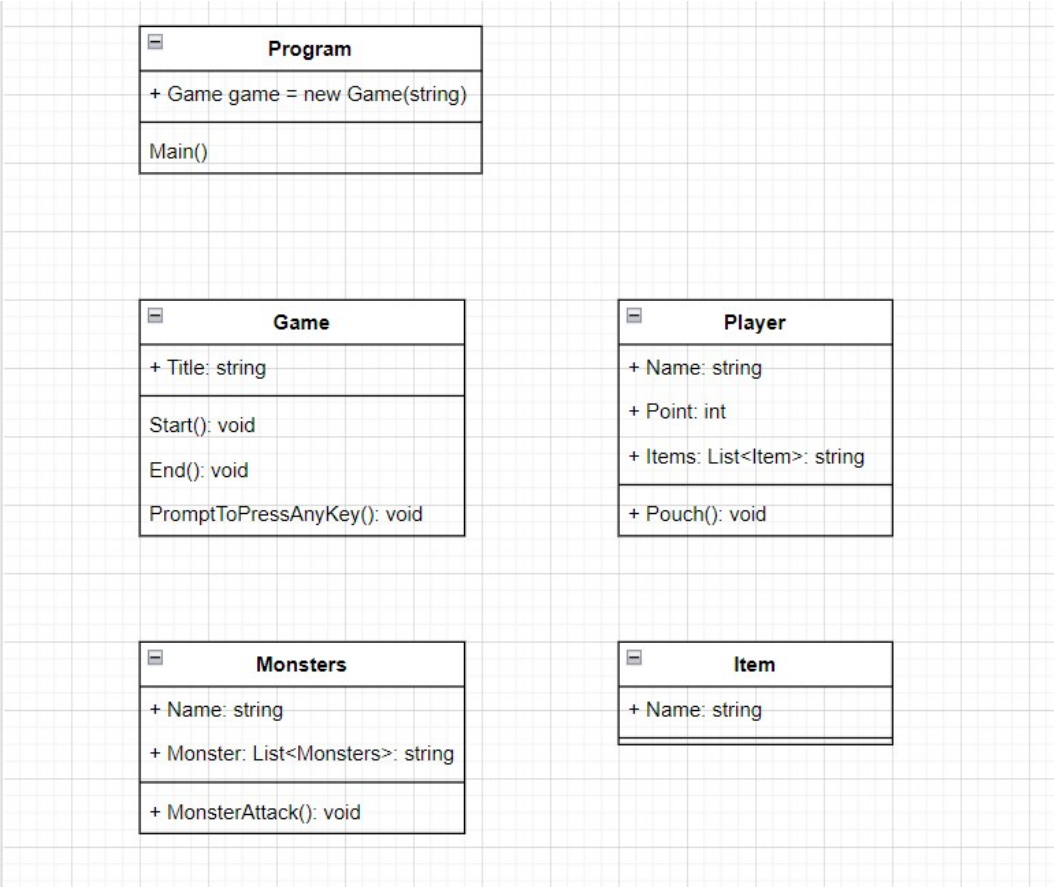
Winning conditions:

- Read the instructions carefully, so you won't choose the wrong option, which leads to game over.
- When encountering monsters, you will receive random points in a certain range (pointGain variable in Game Class) depending on what weapon you choose. You can repeat the fight until you reach the total point of 10 or more to win the fight.
-

Three variables that impact game play:

- **quest1** variable in **Act1()** method in **Game** Class: I think this variable impacts the player experience because if they choose the wrong one, the game will end unexpectedly.
- **quest2** variable in **Act1()** method in **Game** Class: I have the same thought as above for this variable as well.
- **UserChoice3** variable in **Act3()** method in **Game** Class: this variable gives you two options, and one of the options gives you hint for the next round.

UML Diagram:



Credits for the ASCII arts (also included in the game):

- Border by dc, https://www.asciiart.eu/art-and-design/borders
- Little fish by jgs, https://www.oocities.org/spunk1111/aquatic.htm
- Jelly fish by unknown, https://www.oocities.org/spunk1111/aquatic.htm
- Explosive by unknown, https://www.asciiart.eu/weapons/explosives
- Fireworks by jgs, https://www.asciiart.eu/holiday-and-events/fireworks
- Moray eel by jgs, https://www.oocities.org/spunk1111/aquatic.htm
- Shark by jgs, https://www.oocities.org/spunk1111/aquatic.htm
- Swordfish by unknown, https://ascii.co.uk/art/swordfish
- I also got help from one of the tutors Ciarenn Hollis from the IAM department. He helped me with the code to get the total point after defeating monsters:

```
int pointGain = 0;

player.Point = player.Point + pointGain;
```

And the code for displaying items in the list:

```
public void DisplayItem()

{
```

```
        // Display items in the pouch
        int amountOfItem = player.Items.Count;
        for (int i = 0; i <= amountOfItem - 1; i++)
        {
                WriteLine($"{i + 1}) {player.Items[i].Name}");
         }
         WriteLine($"You have {amountOfItem} items");
}
```

In my code, inheritance is shown where method Pouch() is defined in Player Class, and Game Class inherits from Player Class. Method Pouch() was called as an instance of Game Class.

Encapsulation is used when you want to hide data or behaviors that are not useful to the user. There are some access modifiers that you can use like private or protected. In my code, I used mostly public because I feel like there's no data that can be hidden.

Polymorphism has a similar idea to inheritance, but you can use those methods to perform different tasks. For example, in my code, method Pouch() was a list and was called in Game Class. In that Game Class, the components were removed out of the list in method Pouch().