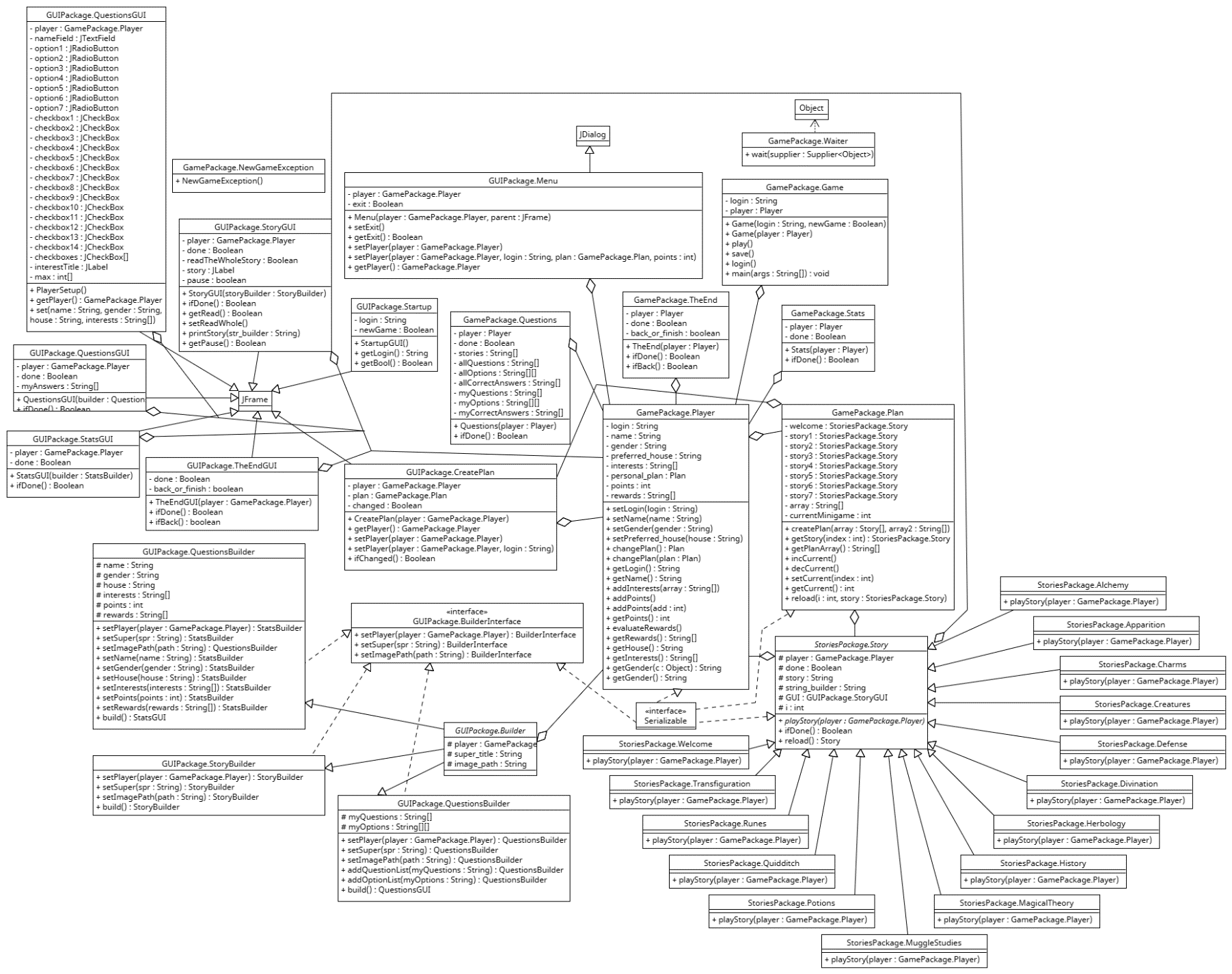
**Open Door’s Day at Hogwarts**

1. **Introduction**
2. **Diagram**
3. **Evaluation criteria**
   1. **Main**

In my opinion, I fulfilled the main criteria of the project. My project meets the given topic and, for the most part, with some changes, meets my objective.

The project also contains inheritance in two separate hierarchies. The first ones are extended from the abstract ‘Story’ class and the second ones are builders extended from the ‘Builder’ class. When it comes to the builders, I also used an interface there. I used polymorphism in many classes, for instance in the ‘Player’ class ‘addPoints()’ and ‘addPoints(int add)’ where in the first case the points are simply incremented and in the second case there is a specific amount added. Every class where it is necessary is encapsulated correctly. The aggregation is used throughout virtually the whole project.

I believe that my code is correctly organized, and I believe that the comments I wrote are more than enough.

* 1. **Secondary**

I believe that I fulfilled 8 of the secondary criteria:

1. Builder design pattern in ‘GUIPackage’ – classes ‘BuilderInterface’, ‘Builder’, ‘StoryBuilder’, ‘QuestionsBuilder’ and ‘StatsBuilder’ – used in e.g. ‘StoriesPackage.Charms’
2. Try/catch exception in ‘GamePackage’ – class ‘NewGameException’ – used in ‘Game’
3. GUI – which counts as 2 criteria - ‘GUIPackage’
4. Multithreading in ‘StoriesPackage’ in every class that extends ‘Story’
5. RTTI in ‘StoriesPackage’ – class ‘Story’
6. Lambda expressions in every ifDone() method and more
7. Serialization
8. **GitHub**