**C Sc 335 Analysis and Design Artifacts for Final Project**

*This must be added to your private Github repo in a directory named documents*

**1. Team Name:** \_\_Risky Business\_\_\_ **Project Manager:** Theo

**2. Team Members**: \_\_Abigail Dodd, Sydney Komro, Dylan Tobia, Jewell Finder

**3. Candidate Objects or Class Hierarchies**

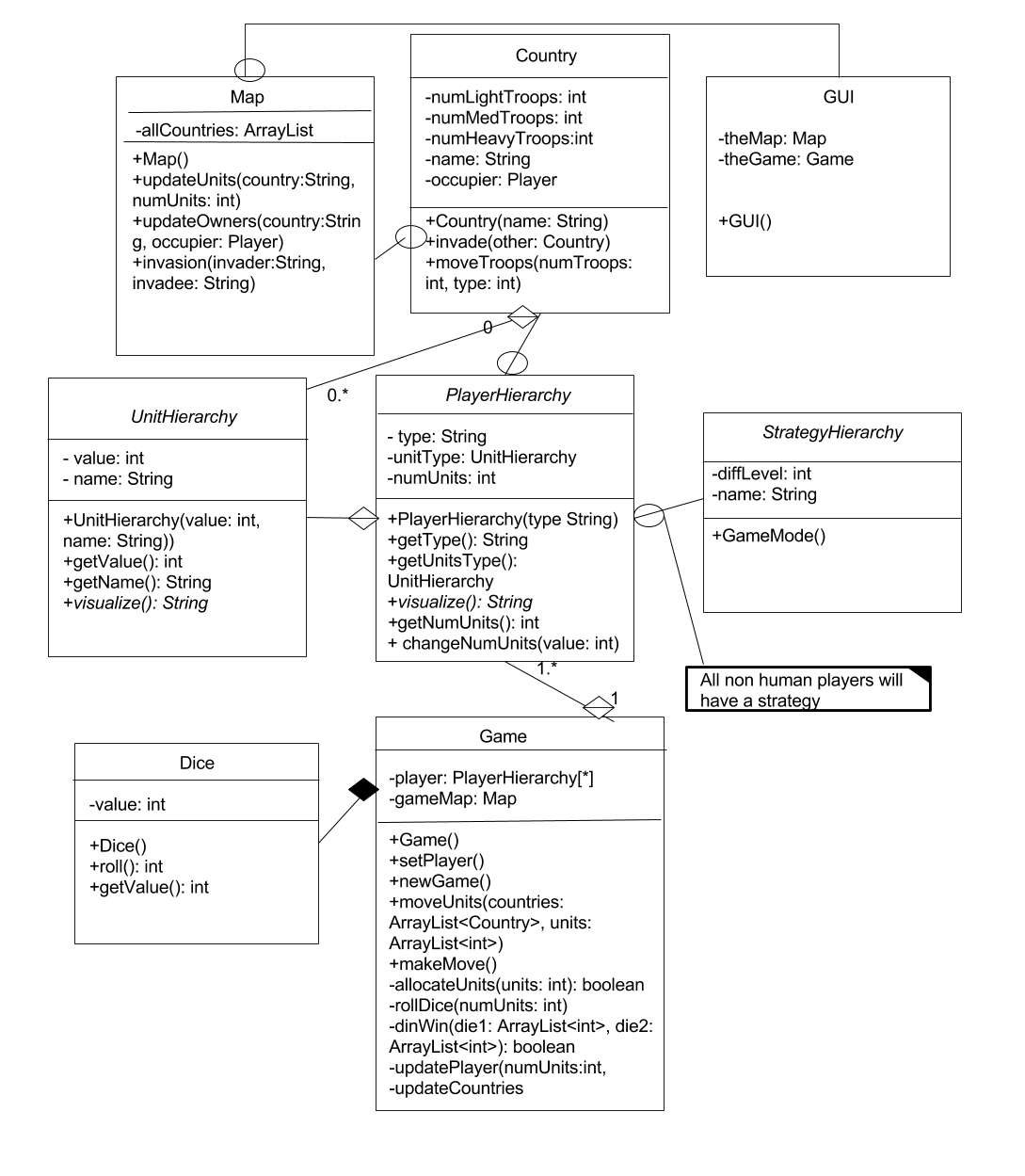
List the most important objects, or the name of an inheritance hierarchy, and the main responsibility.

|  |  |
| --- | --- |
| **Candidate Object** | **Single Responsibility in 1 or 2 sentences** |
| Map | Stores and manipulates the game board |
| Country | Stores country information, including occupier, troops, neighbors |
| Unit hierarchy | Each member of this inheritance hierarchy represents a different type of game unit (i.e. infantry, cavalry, etc.) |
| Player hierarchy | Each player in the hierarchy will represent a different faction that they play as (it’s a hierarchy to allow players to choose from different identities) |
| Strategy hierarchy | Each different ai strategy would be a unique member of this hierarchy |
| Dice | Represents the dice that are rolled at each player’s turn |
| Game | Coordinates and manages the game of risk |
| GUI | Displays the game in a easy to follow manner |
| Player | Represents individual players during the game |
|  |  |

**4. Class Diagram:** Your team UML Class Diagram must show at least all of your candidate objects from above. Show any relationships between them the classes such as inheritance or interface implementation. Draw general associations such as dependency or aggregation. Label some to help explain things. Add any multiplicity adornments that seem appropriate. Use notes to explain things if you feel it will help. Each UML class must show the class name. For full credit, each class must have an average of at least one attribute per class. There must be an average of at least 2.0 methods per class, which may be implicit (no need to repeat methods) if the class implements a Java interface with methods shown there.

*This class diagram may be written by hand and scanned or drawn with any*

*UML editor or drawing program*



**5. Sequence Diagram:** Your team UML Sequence Diagram should show the most important scenario you can think of. Your sequence diagram should show most of your objects from above and how they communicate with each other.

*This Sequence Diagram may be written by hand and scanned or drawn with any program or sequence diagram editor such as*[https://www.websequencediagrams.com/#](https://www.websequencediagrams.com/)

