**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:**  \_\_\_\_\_Riskier Business\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**2. Team Members**: \_\_\_\_\_Colin Widner\_\_\_\_\_\_ \_\_\_\_\_\_Qiming Wan\_\_\_\_\_\_\_\_

\_\_\_\_\_\_Yiling Ding\_\_\_\_\_\_ \_\_\_\_\_\_Mingjun Zhou\_\_\_\_\_\_\_\_

**3. Candidate Objects or Class Hierarchies**

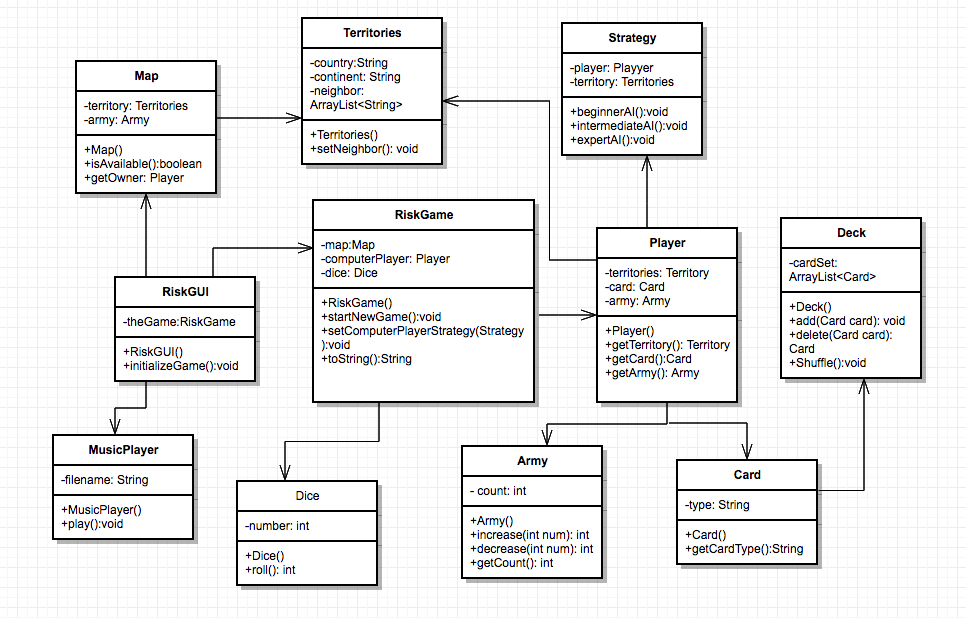
List the most important objects, or the name of an inheritance hierarchy, and the main responsibility.

|  |  |
| --- | --- |
| **Candidate Object** | **Single Responsibility in 1 or 2 sentences** |
| Map | Holds all the armies and coordinates actions. |
| Strategy | AI with different difficulty levels to play. |
| Dice | Randomly roll to determine winner. |
| Territories | Keeps track of armies in territories and other territories it touches. |
| Card | Awarded for capturing territories. Can be turned in for more armies. |
| Army | Game pieces to be put on the map. |
| Player | Holds all territories captured, armies owned, and cards. |
| MusicPlayer | Plays the background music of the game. |
| RiskGUI | Visualizes the game for the user. |
| RiskGame | Holds up one round game information. |
| Deck | Holds all the Risk Cards. |

**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/)

