**Refactoring Document**

1) Modules Moved

* The main function for reinforcement(getReinforcedMap( )) was moved from RiskReinforcementPhase.java controller to the RiskPlayer.java model.
* The main function for attack(getAttackphaseMap ( )) was moved from RiskAttackPhase.java controller to the RiskPlayer.java model.
* The main function for fortification(getFortifiedMap( )) was moved from RiskFortificationPhase.java controller to the RiskPlayer.java model.

This was done to implement reinforcement, attack and fortification as methods of the player class.

2) Modules Added

* In RiskGameHelper.java class, the below modules were added to accommodate the risk rules:
* calculateDominationMapControlled – This method gives the percentage of the map controlled by player.
* getRiskTerritoryByName – This method finds the territory by name to use an attack phase.
* updateArmyAfterAttack – This method updates the army after attack and deletes defender’s territory if necessary.
* moveArmyAfterAttack – This method moves the armies after every attack.
* assignedRandomCard – This method assigns random cards to the player after winning a territory.

3) Refactoring of game code and JUnit test cases to accommodate the above mentioned modules changed and module added.

* Code level changes/additions were done to accommodate the changes made while implementing the observer design pattern for the various phases in the game.