**Refactoring & Design Pattern Document**

**Note:** Here are the list of refactoring because we did not comment on all the commits on GitHub. We have refactored the following list in order to follow observer pattern and Model View Controller.

**Design Patterns:**

**1) Observer Pattern:**

GameFacade is the main class that has a Game, User, UserProfile and Invitation object. Everything from the UI goes through the GameFacade class.

**2) Model View Controller:**

We have some controllers in the UI Package which is integrating GameFacade with the UI. InvitationPanelController,

ActiveGameController, ProfileController, and GameController all require database access so they are handled by GameFacade.

**Refactoring :**

1. Moved navigation buttons to supperclass called MainPanel.java
2. Removed ChessBoard object from GameFacade and accessed it through the Game class in order to follow MVC
3. Refactored Duplicate Code to a method called outerRingValid**Side**Moves() and outterRingValid**Next**Moves() in Rook.java class
4. Broken validMoves() method to two separate method called outerRingValid**Side**Moves and outterRingValid**Next**Moves() in Rook.java class
5. Refactored unused global variable in all Classes
6. Refectored ClientUI to UI package
7. Refactored code to Business Layer and Foundation packages
8. Refactored DropDown Menu of active game of identifier of GameId to “nickname” of user
9. Refactored method name to make it more clear

10.Removed SerializedgameTest.java from Business Layer package

11.Removed Test.java from Business Layer package which was for CLI testing

12.Refactored print statement with printStackTrace() method in all class

13.Refactored System.exit(1) from all classes

14.Created dummy functions in order to follow Observer pattern

15.Removed Main() method from classes

16.Refactored removelegalMove() and add helper function

17.Refactored message on Loginpanel

18.Refactord query syntext:instead of setting is\_active is zero now it will remove a row from table

19.Refactored success/failure message of send invitation in invitationpanelController.java

20.Refactored promotion() added condition for checking color

21.Refactored MainPanel:removed unnecessary messages

22.Refactored move() signature: now returns String

23.Added dummy methods called getValidMoves(),getType(),move() in game class to follow observer pattern

24.Refactored GamePanel.java added dropdown to show list of active games associate with user

25.Added lastUpdated column in gameserialized table

26.Refactored move():now it will update state when user make move()

27.Added Color attributes in getNextForward() ,getNextDiagonal(), getSideways() and getprevBackward() , getpreviousDiagonal() in ChessBoard.java

28.Changed Highlighted Move color in GamePanel.java

29.Move listActivegame() from Game.java to User.java