**CS151 Group Project Final Report**

**Team Name:** Ocean

**Team members:** Steven Guan, Shawn Darrel Suarez, and Ryan Woo

1. **Write Up for Design Pattern Assessment // Shawn could you help out to complete**
   * 1. Write the NAME of one of the controller classes (or a class that contains a controller). Copy and paste a code segment of the controller that calls the mutator of the model
     2. Write the NAME of the model class. Copy and paste a code segment of a mutator of the model that modifies data and also notifies view(s). Give me the name of mutator as well.
     3. Write the NAME of a view class. Copy and paste a code the notification method of the view and show me how the notification method paints the view using the data from the model.
     4. Write the NAME of a strategy and copy the code.
     5. Write the name of two concrete strategies. (Just names required.)
     6. Copy and paste the code segment where you create a concrete strategy and plugin into the context program.
2. **Class diagram (simple class diagram)**



1. **Use cases**

***User Case: Choosing the style of board with number of stones***

|  |  |  |
| --- | --- | --- |
| Step | User’s Action | System’s Response |
| 1 | User opens the program. |  |
| 2 |  | System display empty main board and option to choose board style with number of stones  Style Option: Simple or Fancy  Stone Option: 3 or 4 |
| 3 | User select board from drop down menu |  |
| 4 |  | System display selected board and stones per user’s input. |
| 5 |  | Main board display Current player, undo info (allows 3) and player’s scores. |

***User Case: Play Game***

|  |  |  |
| --- | --- | --- |
| Step | User’s Action | System’s Response |
| 1 | Player 1 pick a pit to start a game |  |
| 2 |  | System run the board that move stones from the selected pit and insert one stone in each pit counter-clock wise. |
| 3 |  | System always start with player 1 and end player 1 if player 1 click button End Turn at any time regardless free turn when the last stone drops into own Mancala. System automatically collect stones from the other player when the last stone drop into an empty pit at player own side and system collect stones to player’s Mancala |
| 4 | Player 2 select pit to collect on his side to start his turn. |  |
| 5 |  | System end player 2 if player 2 click button End Turn at any time regardless free turn when the last stone drops into own Mancala. System automatically collect stones from the other player when the last stone drop into an empty pit at player own side and system collect stones to player’s Mancala |
| 6 |  | System will automatically ends the game when pits are empty on both player’s side and system count scores / declare tie or winner. |

Variation #1

1. In step 1, player 1 clicks opponent side of pit.
2. System responses:

Nothing happens (Assume player knows the rule of Mancala)

1. Continue with step 1.

Variation #2

1. In step 2, if player 1 selected a pit and last stone drops into the Mancala of player 1,

then, option to end the turn or select another pit and end the turn

1. Continue with step 2.

Variation #3

1. In step 3, player 2 clicks opponent side of pit.
2. System responses:

Nothing happens (Assume player knows the rule of Mancala)

1. Continue with step 3.

Variation #4

1. In step 4, if last stone drops into the Mancala of player either player 1 or 2. Stones collects.
2. Continue with step 4.

Variation #5: Undo

1. Either player click undo button whenever pit are selected, system back to prior status
2. Number of undo are updated on the main board (each player are allowed 3 undo).

Variation #6: Settings

1. Players click settings button any time during the game to change the style of the board
2. **Sequence diagram:**

Steven Guan

1. **One page of paper that includes answers for the following questions**
   1. Which materials/key concepts from this course did you apply on the project?

MVC pattern and strategy pattern are applied in this project as required per this project. Objects are designed with interface classes and regular classes that each of them are communicated upon is-a, has-a, etc. relationships. Implementation of GUI is a big part of this project that required knowledge of how to use the MouseAdapter and ActionListeners as well as Jbutton, JPanel, etc. Methods are implemented on the structure of the game therefore in order to achieve the functionalities of this project, we required to spend significant amount of time to develop the data structures. Regardless of Mancala game easy algorithm, it was hard to connect the dot meaning communication between classes (interface) and extends/implements required JComponents and ChangListeners.

* 1. Which topics did you have to learn through self-study in order to complete the project?

Despite the knowledges from this course, for instance, MVC or Strategy pattern, applying model, view, controller can be challenging if certain requirements to be met. When program is not working as intended, we required to review from the chapters or online sources, for example the use of GUI in term of Graphics2D to draw shape, or MouseAdapter to use MouseClicked, or layout in which all pieces comes together.

Team, please read, add more or correct if necessary