**PROJECT GAME DESIGN**

**Satyam Khare**

**email:satyam.khare@flipkart.com**

**SECTION I: First Iteration Game Project Design**

**[Till Phase III or Phase IV Submitted Upto 29th January, 2020]**

**PART I : Class Diagrams:**

**Open the following link to view Class Diagrams associated with the project**

:

<https://drive.google.com/open?id=1_b2aMfwCYHERuyNNEAtvU6l-U5XAZ-BJ>

**PART II : Common Design/Choices and Conventions/Assumptions and Detailed Descriptions etc.**

GameManager is an interface which consists of method playGame(),this allows the project to choose any game,if in future need arises.

Since during first iteration of the project,task of project was to develop the tic-tac-toe game to be played on 3\*3 either between two human players or against the computer.

Therefore i introduced the interface TicTacToeGameModes which consists of method playGame(),and TicTacToeGameManager class which implements GameManager interface.Based on the choice of inputs given by user,TicTacToeGameManager runs the game with the corresponding game mode.

Currently the game run on two game mode in first design,3\*3 against computer and 3\*3 against human player.There is a separate method which decides how computer should play the particular move,therefore the algorithm can be easily changed at any time.

**PART III : Feature Specific Design/Choices and Conventions/Assumptions**

**GameDesign v2.0 - Requirement I**

**<STATUS> - 1. Tic-Tac-Toe consists of 3x3 Square Cells**

Feature Specific Design Decision?

Completed

**<STATUS> - 2. Game Between Two Humans**

Feature Specific Design Decision?

Completed

**<STATUS> - 3. Game Between Human and Machine**

Feature Specific Design Decision?

Completed

**<STATUS> - 4. Winning Criteria - 3 Cells in Row/Column/Diagonal are in Same State.**

Feature Specific Design Decision?

TicTacToeRules class is created which consists of methods which define where TicTacToe game is won or in a draw state.

Completed

**<STATUS> - 5. Announce Winning Player**

Feature Specific Design Decision?

Completed

At the start of the game,players are required to input their names,and the winner name gets displayed at the end of the game.

GameDesign v2.0 - Requirement II

**<STATUS> - 6. Enhanced Tic-Tac-Toe Game Consist of 9x9 Squares…**

Additional GameMode was introduced which implements the interface GameMode to incorportate 9\*9 TicTacToeGame against computer as well as human player.

Feature Specific Design Decision?

**<STATUS> - 7. Enhanced Tic-Tac-Toe will continue to expand in depth levels...**

Feature Specific Design Decision?

There is a class named TicTacToeRules,which consists of method defining winning criteria and draw criteria.Before the above mentioned design ,there are separated winning and draw criteria methods of 3\*3 TicTactoe mode and 9\*9 TicTacToe mode,but to incorporated much more levels,these methods are made generic and can accept any TicTacToe of size n\*n where n is some power of 3.This allows the game to be played at any level of depth.

**<STATUS> - 8. Extend Game to 4x4 Board**

Feature Specific Design Decision?

Completed.

Additional mode TicTacToe 4\*4 is introduced which implements TicTacToeGameModes interface.

**<STATUS> - 9. Human Player is Biased...**

Feature Specific Design Decision?

Completed

Last position of both the players is saved ,so if the player decides to undo the last move,they can easily do so,using the undo option.

**<STATUS> - 10. Storing and Retrieving Game State**

Feature Specific Design Decision?

Completed

**<STATUS> - 11. Store Players Game Statistics: Leaderboard**

**Feature Specific Design Decision?**

Completed

Whenever a new game starts,players are required to input their names,and whenever the player wins the game,their statistics are stored in the hashmap which consists of key as the player name,and value as the number of wins.

GameDesign v3.0 - Requirement III

**<STATUS> - 12. Super Tic-Tac-Toe Game Extends Enhanced Tic-Tac-Toe Game…**

Done

Feature Specific Design Decision?

**<STATUS> - 13. Design Winning and Losing Criterias On All Edges...**

Feature Specific Design Decision?

**<STATUS> - 14. Incorporate Irregular shaped Hexagonal Boards**

Feature Specific Design Decision?

GameDesign v4.0 - Requirement IV

**<STATUS> - 15. Incorporate Biased Game Board**

Feature Specific Design Decision?

**<STATUS> - 16.Incorporate Connect Four Game In Design**

Feature Specific Design Decision?

**<STATUS> - 17. Discover Newer Abstract Types**

Feature Specific Design Decision?

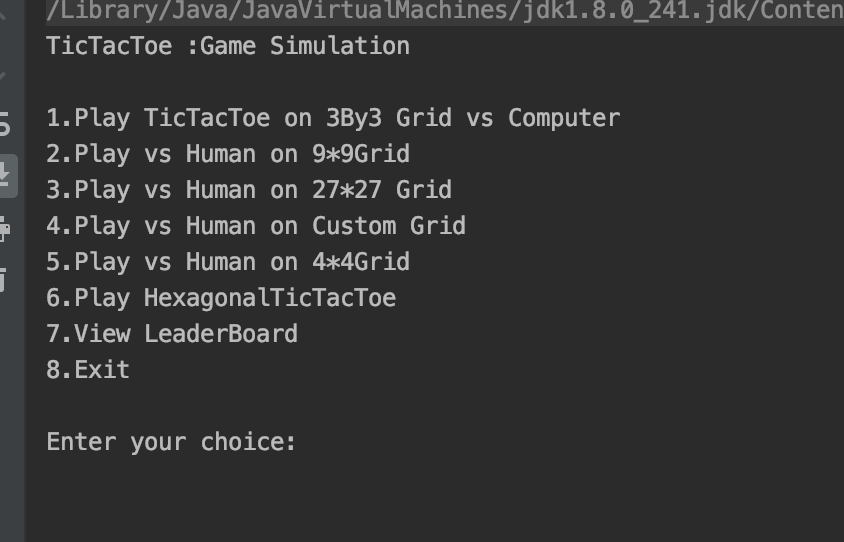
**<STATUS> - 18. Refactor and Reuse Code In Both Games**

Feature Specific Design Decision?

Completed

**SECTION II: Second Iteration[Refactoring/Redesign] Game Project Design**

**[Till Phase III or Phase IV Submitted Upto 03rd February, 2020]**

****

**PART I : Common Design/Choices, Conventions and Assumptions**

**Sequence Diagrams**

**Class Diagrams**

**Dependency Diagrams**

**Class diagrams Link:**

<https://drive.google.com/open?id=1_b2aMfwCYHERuyNNEAtvU6l-U5XAZ-BJ>

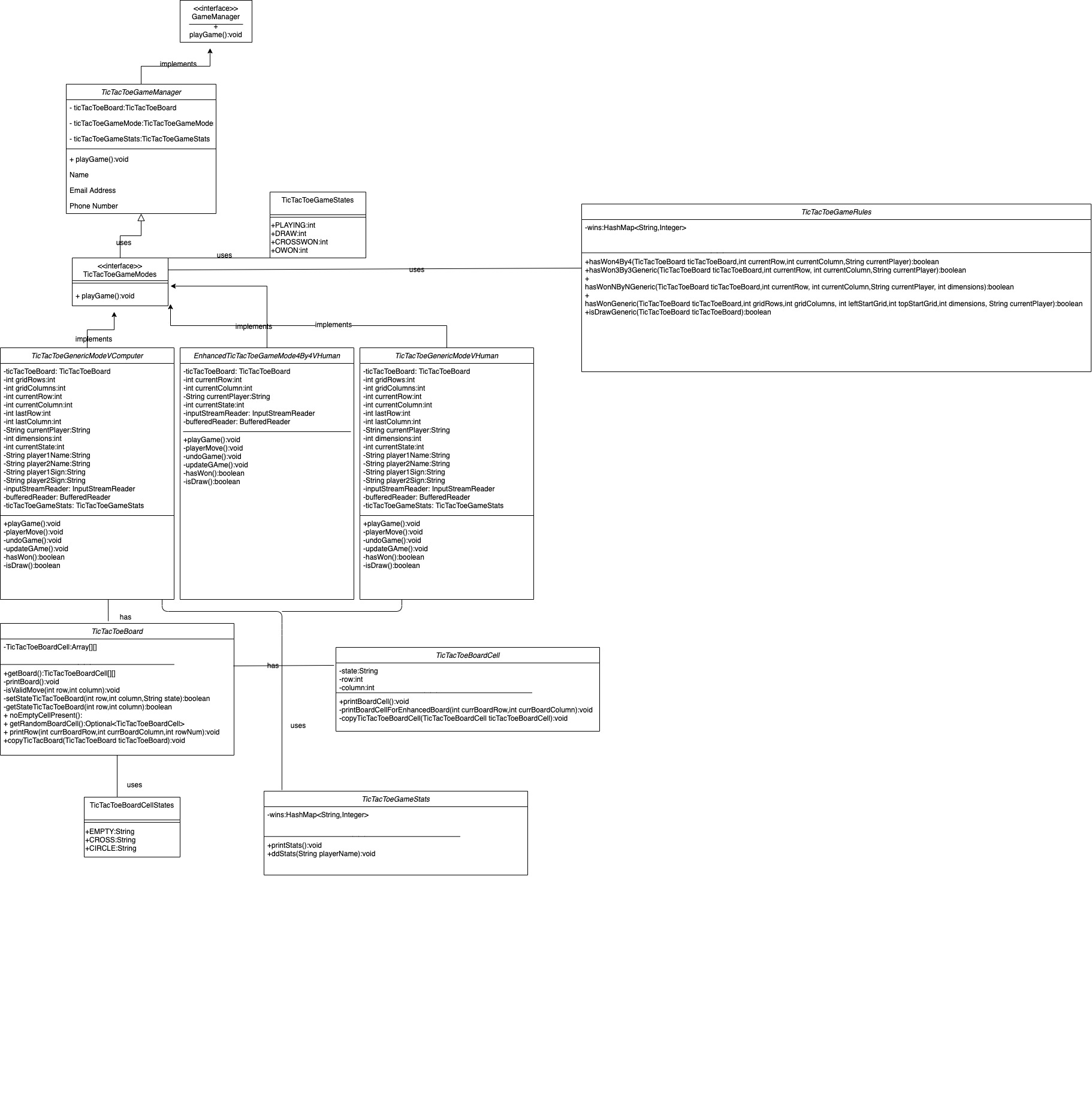
Else the image is included in document as well.

**PART II : Common Design/Choices and Conventions/Assumptions and Detailed Descriptions etc.**

Code Refactoring is done,and updated class diagrams are provided in the design document.

Addition of class which includes static final Strings for TicTacToe Game states and TicTacToeGameModes.

TicTacToe Enhanced game is made generic in class TicTacToe GameRules,and now the game can be played at any depth levels of n\*n.



**PART III: Feature Specific Design/Choices, Conventions and Assumptions**

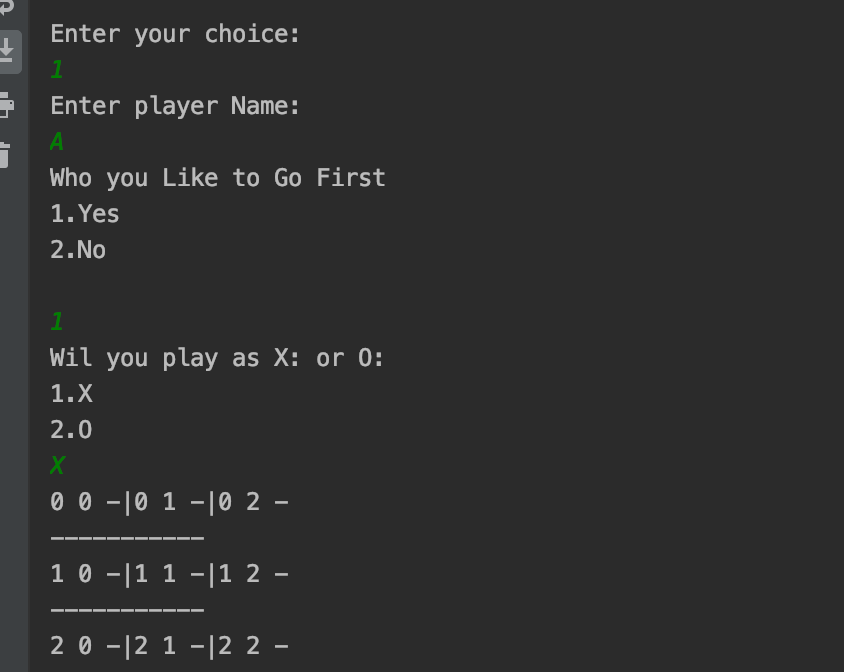
**GameDesign v2.0 - Requirement I**

**<STATUS> - 1. Tic-Tac-Toe consists of 3x3 Square Cells**

Feature Specific Design Decision?

Completed

TicToeBoard class is responsible for the TicTacToe board ,which consists of TicTacToe Cells.



**<STATUS> - 2. Game Between Two Humans**

Feature Specific Design Decision?

Completed

Game between two humans is extended to be played on a n\*n TicTacToe board.

<STATUS> - 3. Game Between Human and Machine

Feature Specific Design Decision?

Completed

Game between human and machine is extended to be played on a n\*n TicTacToe board.

**<STATUS> - 4. Winning Criteria - 3 Cells in Row/Column/Diagonal are in Same State.**

Feature Specific Design Decision?

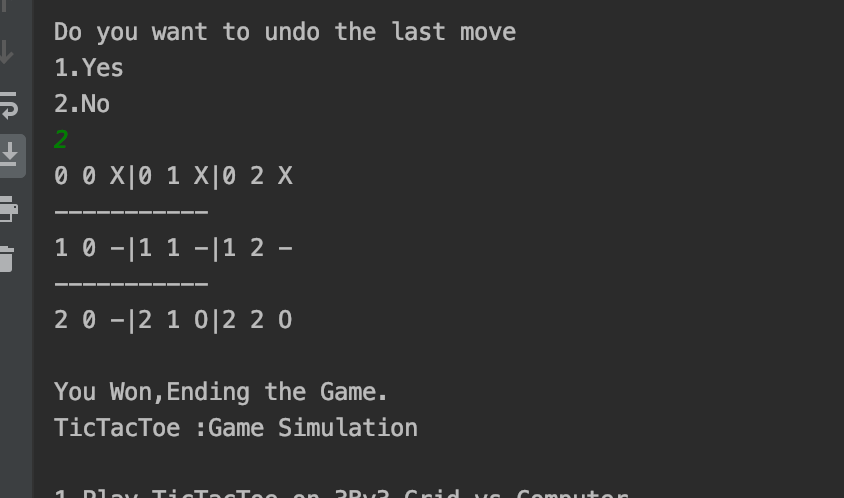
Completed

**<STATUS> - 5. Announce Winning Player**

Feature Specific Design Decision?

GameDesign v2.0 - Requirement II

Completed



**<STATUS> - 6. Enhanced Tic-Tac-Toe Game Consist of 9x9 Squares...**

Feature Specific Design Decision?

Completed

**<STATUS> - 7. Enhanced Tic-Tac-Toe will continue to expand in depth levels...**

Feature Specific Design Decision?

Completed

Generic method are introduced

hasWon3By3Generic(TicTacToeBoard ticTacToeBoard,int currentRow,

int currentColumn,String currentPlayer)

And

hasWonNByNGeneric(TicTacToeBoard ticTacToeBoard,int currentRow,

int currentColumn,String currentPlayer,

int dimensions)

Which evaluates when the game is in winning state or not.

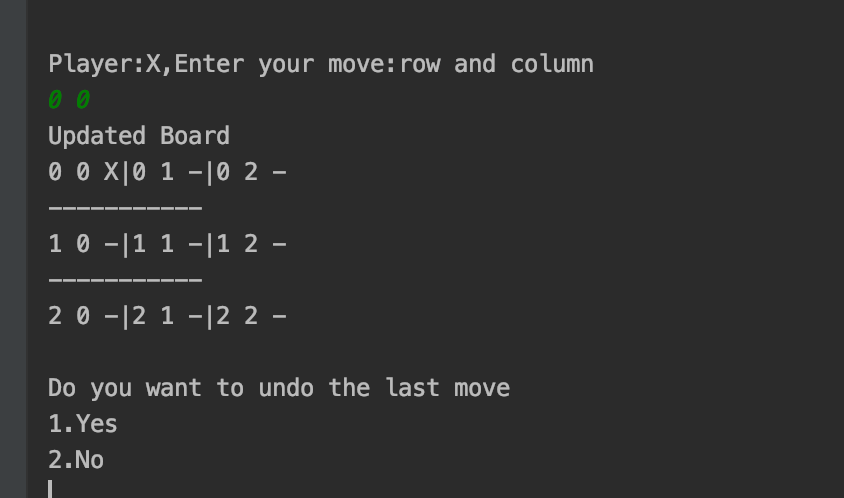
**<STATUS> - 8. Extend Game to 4x4 Board**

Feature Specific Design Decision?

Completed

**<STATUS> - 9. Human Player is Biased...**

Feature Specific Design Decision?



Completed

Undo method is extended for enhanced TicTacToe Game mode as well.

**<STATUS> - 10. Storing and Retrieving Game State**

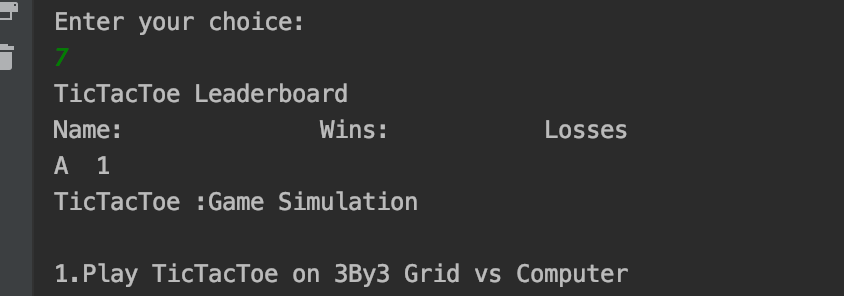
Feature Specific Design Decision?

Completed

**<STATUS> - 11. Store Players Game Statistics: Leaderboard**

Feature Specific Design Decision?

Completed

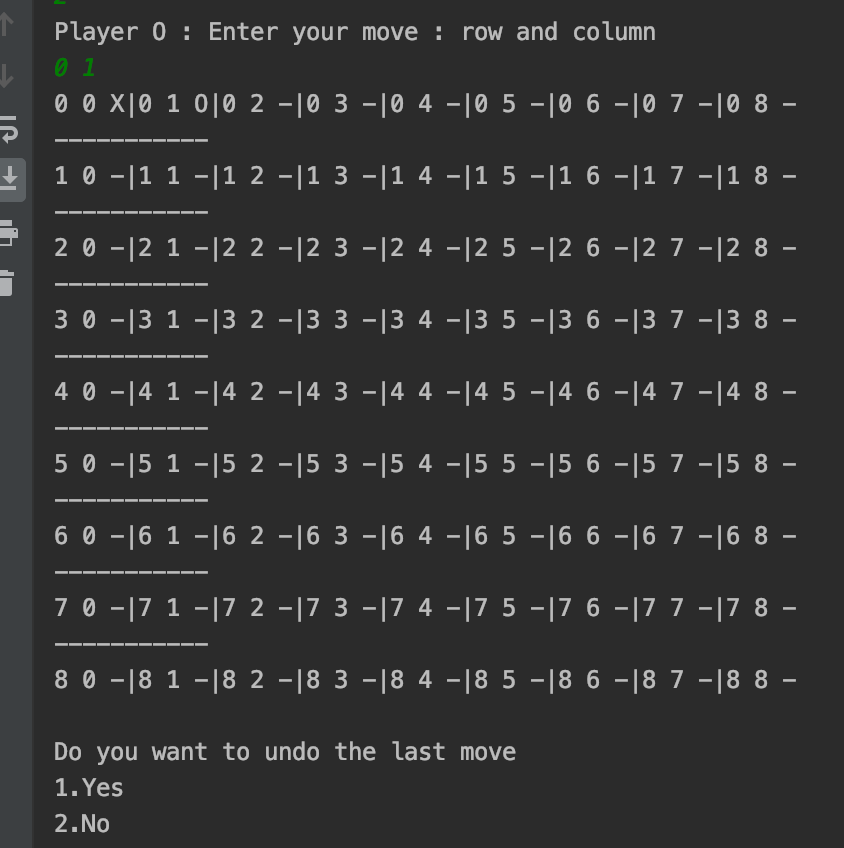


**GameDesign v3.0 - Requirement III**

**<STATUS> - 12. Super Tic-Tac-Toe Game Extends Enhanced Tic-Tac-Toe Game...**

Feature Specific Design Decision?

Completed



**<STATUS> - 13. Design Winning and Losing Criterias On All Edges...**

Feature Specific Design Decision?

**<STATUS> - 14. Incorporate Irregular shaped Hexagonal Boards**

Feature Specific Design Decision?

**GameDesign v4.0 - Requirement IV**

**<STATUS> - 15. Incorporate Biased Game Board**

Feature Specific Design Decision?

Completed

**<STATUS> - 16.Incorporate Connect Four Game In Design**

Feature Specific Design Decision?

Completed

**<STATUS> - 17. Discover Newer Abstract Types**

Feature Specific Design Decision?

Completed

**<STATUS> - 18. Refactor and Reuse Code In Both Games**

Feature Specific Design Decision?

Completed

**SECTION III:**

**How to Run/Test Your Code?**

**Describe How To Run Your Code**

Project is built using maven build tool.

Checkout the dev branch and run the maven project.

Starting point is Main.java

**Are you providing all Input/Output files to run Test Code using Test.java?**

No other files are required.Maven imports all the necessary libraries.

**I have all you followed following guidelines given for writing test cases, if not then describe it what is working and what is not.**