James Nguyen

Professor Di Troia

CS 151, Section 06

December 15, 2019

Tic Tac Toe Analysis and Design

Analysis of Tic Tac Toe Application

1) Functional Specifications

The Tic Tac Toe Application is an application in which users are able to play a game of Tic Tac Toe against a computer as well as observe past games that have been played. The application has many different specific functional requirements that should be met.

The application should follow the rules of Tic Tac Toe when a user is playing the game. The game should be played by two players, the computer and user, in which each player will be given a X or O depending on the order they're going. The player that is going first should get an X and the second player should get an O. The game should allow players to take turns placing their symbol in a 3x3 board with each column having 3 boxes and each row having 3 boxes. The application should end the game if either the X's or O's are three in a row, three in a column, or three in a diagonal within the 3x3 board, and announce the winner which is the player with the symbol that won. The application should also end the game once the board is filled up and there are no winners and should announce that it is a tie.

The game within the application should have two levels of difficulties for the computer. The first difficulty should be having the computer place its symbol on a random legal open box. The next difficulty should have the computer blocking any close winning moves by the user, as well as making corner moves in the beginning which are very strategic moves. The user of the application should be able to have the power to change difficulty level.

Within the application, users should be able to view and observe past games that have been played. The application should keep a log of the different games that have been played from the current instance of the application launch as well as previous application launches, and should load all the games as long as the data file that contains the data of the different games are not tampered with. Users should be able to select any game in the past and view the game . The User should then be able to view a game move by move as well as go back and view the previous moves if there are any.

2) User Manual

The Tic Tac Toe Application is an application of Tic Tac Toe in which users can play the game as well as view past games. Upon running the application, the user will be first placed in the Main Menu of the game, on this main menu, there will be two buttons as well as two drop down menus.

The first button will be labeled as “Play Tic Tac Toe” and will be next to a drop-down menu which contains two items, ‘Difficulty 1’, and ‘Difficulty 2’. The selected Difficulty will be used when the ‘Play Tic Tac Toe’ button is clicked. When this button is clicked, the main menu will disappear, and another window will appear containing 9 buttons and a text directing whether the user is the first or second player and their symbol. Now, one can be able to play Tic Tac Toe

with the computer with the selected difficulty by taking turns and clicking on the nine buttons to place the user's symbol in the 9x9 grid. The goal of Tic Tac Toe is to either get the user's symbol to be three in a row, three in a column, or three in a diagonal within the 3x3 board. If an invalid move is made, the application will prompt that it is an invalid move and another move should be made. If the game ends with no winners, the application will prompt the user that the game ended and there is a tie. If either the computer or user wins, the application will prompt the user that game is over and the winner of the game. Once the game has ended, the user has the option to play again, or return to the main menu through two buttons labeled as "Play Again" and "Main Menu".

The second button will be labeled as "View Past Game" and will be next to a drop-down menu which contains a certain number of items labeled as 'Game #' with the number of the game at the end. The number of items in the drop-down menu will depend on the number of games that have been loaded and saved. If there are no past games, then the drop-down menu will be blank. If the button is clicked with an empty drop-down menu, an error message will pop out prompting the user that there are no previous games. Otherwise, the selected game number will be used when the button is clicked and a window will appear with a Tic Tac Toe Board, and three buttons labeled as "Main Menu", "Next", and "Back" will appear on the window as well. The Main Menu button will take the user back to the main menu. The next button will show the next move on the Tic Tac Toe Board that the user would like to view and will be disabled if there are no moves after. The back button will show the pervious move on the Tic Tac Toe Board that the user would like to view, this button will be disabled if there are no other previous moves.

With these buttons and drop-down menus explained, the Tic Tac Toe application can easily be used with ease and free of problems.

3) Use Cases

There are a set of use cases within the Tic Tac Toe Application which will be explained within this section.

Use Case #1: Winning Tic Tac Toe

1. User launches Tic Tac Toe application.
2. The System loads all previous games from the data file, “DATA_TicTacToe.txt”, on the desktop if there are any data.
3. The System opens the Main Menu window.
4. User clicks the “Play Tic Tac Toe” Button with either difficulties selected.
5. The System attains the selected difficulty.
6. The System makes the Main Menu invisible.
7. The System opens the Tic Tac Toe Window, then prompts the User: “You are X, and will be starting first.”
8. The User clicks on a button on the board to make a legal move
9. The System marks the button as ‘X’.
10. The System saves the move into the Moves Saver.
11. The System puts a legal move on the board based on the difficulty chosen, then prompts “Your move.” to the user.
12. The System saves the move into the Moves Saver.
13. Steps 7 to 11 are continued until the User making a winning move.

14. The System disables all buttons and prompts the user “GAME OVER. You are the winner!” and two buttons labeled “Play Again” and “Main Menu” appear on the window.
15. The System saves the moves into a Tic Tac Toe saver, and write the moves into a data file on the desktop labeled “DATA_TicTacToe.txt”.

Variation #1: Invalid Move

1. In Step 8, the User places an invalid move.
2. The System prompts to the user: “Invalid Entry. Choose a space that is empty.”
3. Continue with Step 7.

Variation #2: Computer Wins.

1. In Step 11, the Computer ends up making a winning move.
2. The System saves the move into the Moves Saver.
3. The System disables all buttons and prompts the user “GAME OVER. The Computer is the winner!” and two buttons labeled “Play Again” and “Main Menu” appear on the window.
4. Continue with Step 15.

Variation #3: User Closes Window

1. In Step 13, the user closes the Window.
2. The Moves are discarded and are not saved.

Use Case #2: Tic Tac Toe Tie

1. Steps 1 to 12 from Use Case #1 is performed.

2. The System and User continues to take turns making moves until the board is filled, and there are no winners.
3. The System disables all button and prompts the user “GAME OVER. There are no winners.”, and two buttons labeled “Play Again” and “Main Menu” appear on the window.

Use Case #3: View Past Games

1. User launches Tic Tac Toe application.
2. The System loads all previous games from the data file, “DATA_TicTacToe.txt”, on the desktop if there are any data.
3. The System opens the Main Menu window.
4. User clicks the “View Past Game” Button with a game selected.
5. The System makes the Main Menu window
6. The System attains the selected game and gets the game.
7. The System opens the Tic Tac Toe Window with all the game buttons disabled, the “Main Menu”, “Next”, and “ Back” buttons are placed in the window, and the first move of the game load. The “Back” button is disabled.
8. The User clicks on the Next button.
9. The System shows the next move on the board.
10. Steps 6 and 7 are repeated until there are there are no next moves.
11. The System disables the next button.

Variation #1: No Past Games

1. In step 4 the User clicks the “View Past Game” Button with an empty item selected.

2. The System opens an Error Window which prompts the user: “There are no previous games to view.”

Variation #2: Back Button Pressed

1. In Step 10 , the User presses the back button.
2. The System shows the previous move. If there are no other previous moves, the System disables the back button.
3. Continue with Step 8.

Variation #3: File Does Not Exist

1. In Step 2, the file “DATA_TicTacToe.txt” is unable to be found.
2. The System creates a file with the name “DATA_TicTacToe.txt” on the desktop.
3. Continue with Step 3.

Design of Tic Tac Toe Application

1) Classes and Responsibilities

1. MainMenuWindow: The Main Menu Window will create a graphical user interface in which users are able to play a game of tic tac toe and choose the difficulty of the computer, or view past games and choose the specific game they would like to view. The class will also be connected to a TicTacToe object which will attain the data and perform the necessary operations when an Action Event occurs.

2. TicTacToe: This class will be in charge of the background logical aspects of a Tic Tac Toe game as well as controlling the TicTacToe Window class to update the board that is being used for different operations such as for playing or for viewing a past game.

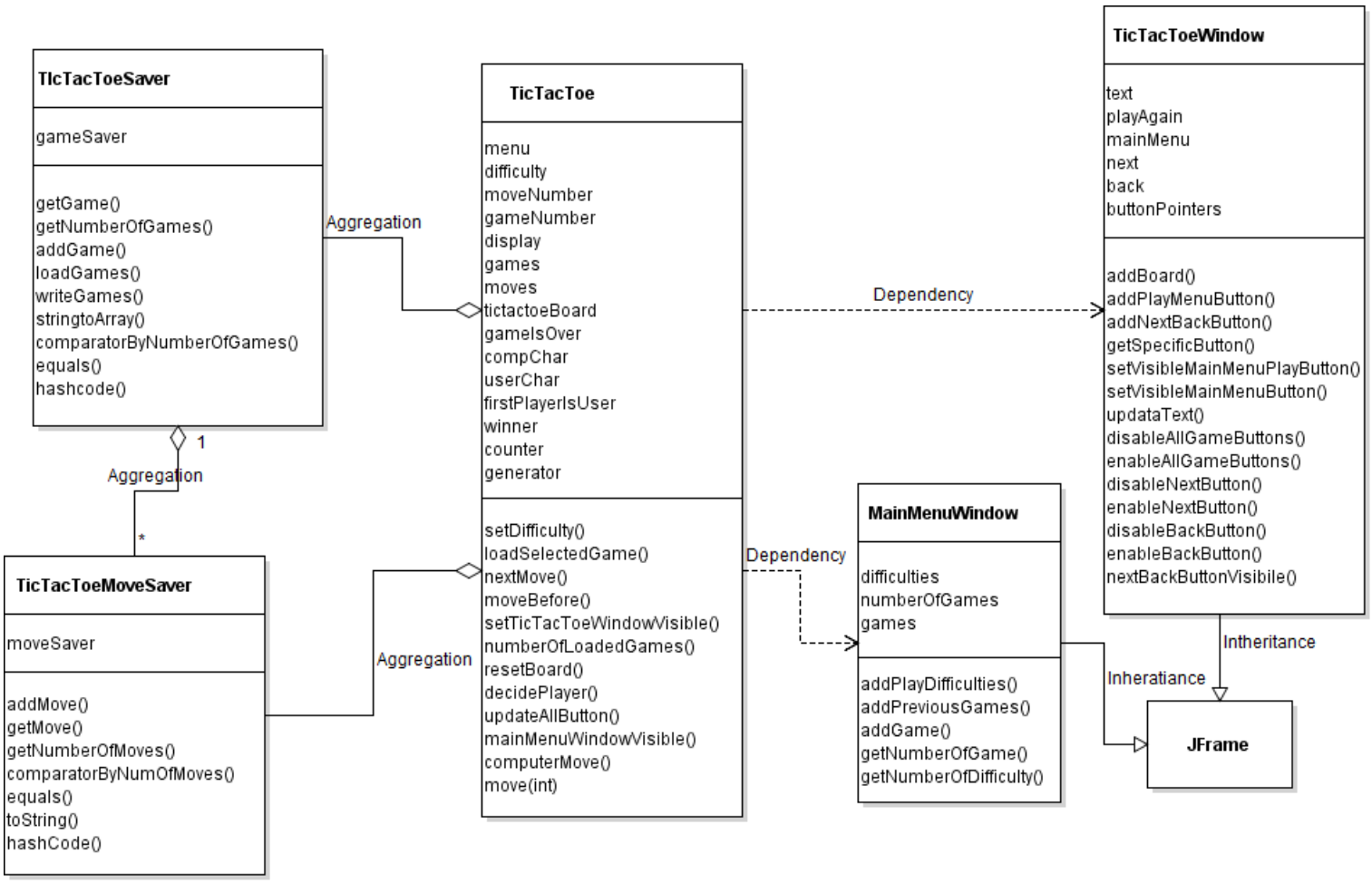
3. TicTacToeMoveSaver: This class will be used to save the different moves of a Tic Tac Toe game through the use an Array List which stores 2D arrays.

4. TicTacToeSaver: This class will be used to save the different games of Tic Tac Toe through the use of an Array List which stores TicTacToeMoveSavers. The class will also contain methods to load data from a text file, "DATA_TicTacToe.txt" on the desktop and turn them into games, as well as write the saved games into the text file.

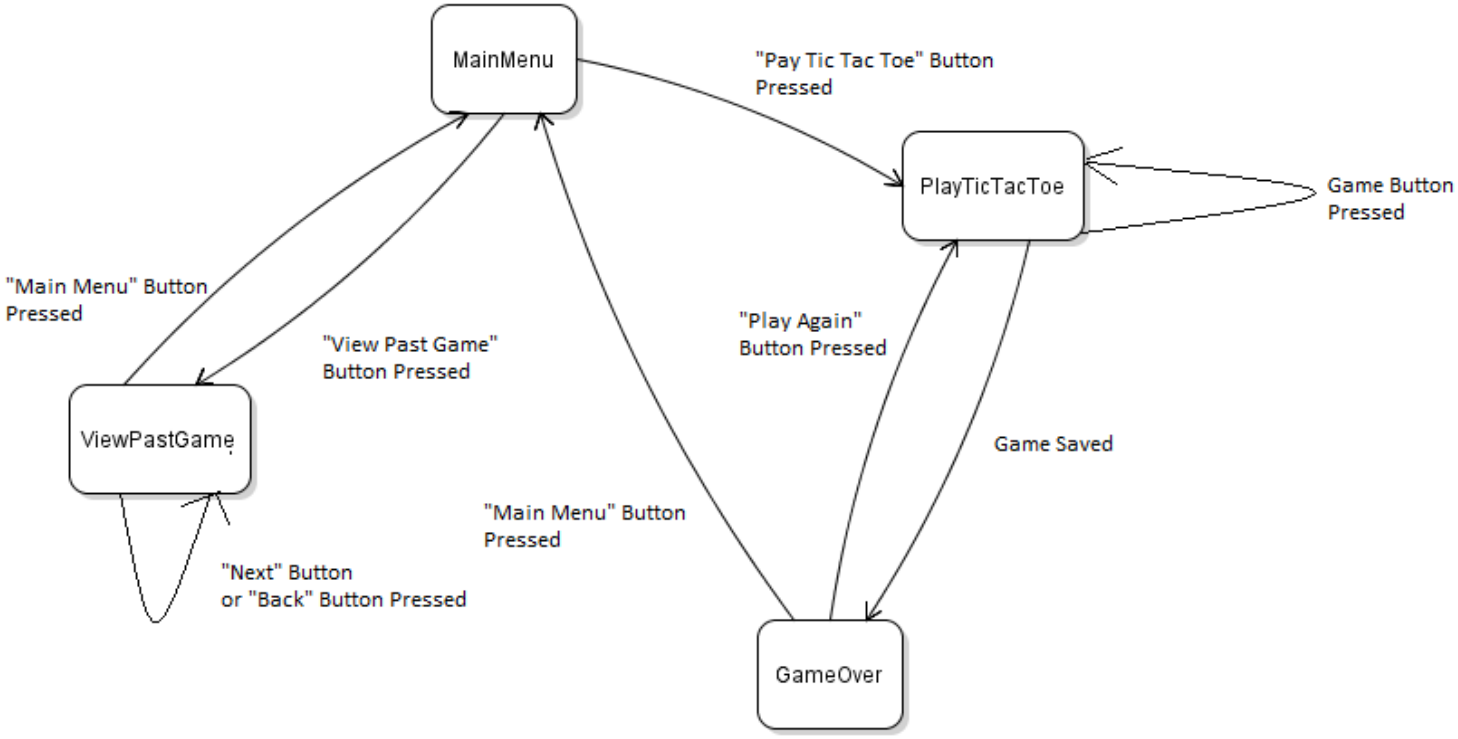
5. TicTacToeWindow: This class will be used to create a graphical user interface in which users are able to interact with a game board through a 3x3 button grid. The class will be controlled by a TicTacToe object which will attain data and perform the necessary operations when an Action Event occurs.

2) UML Diagrams

Class Diagrams and Class Relationships

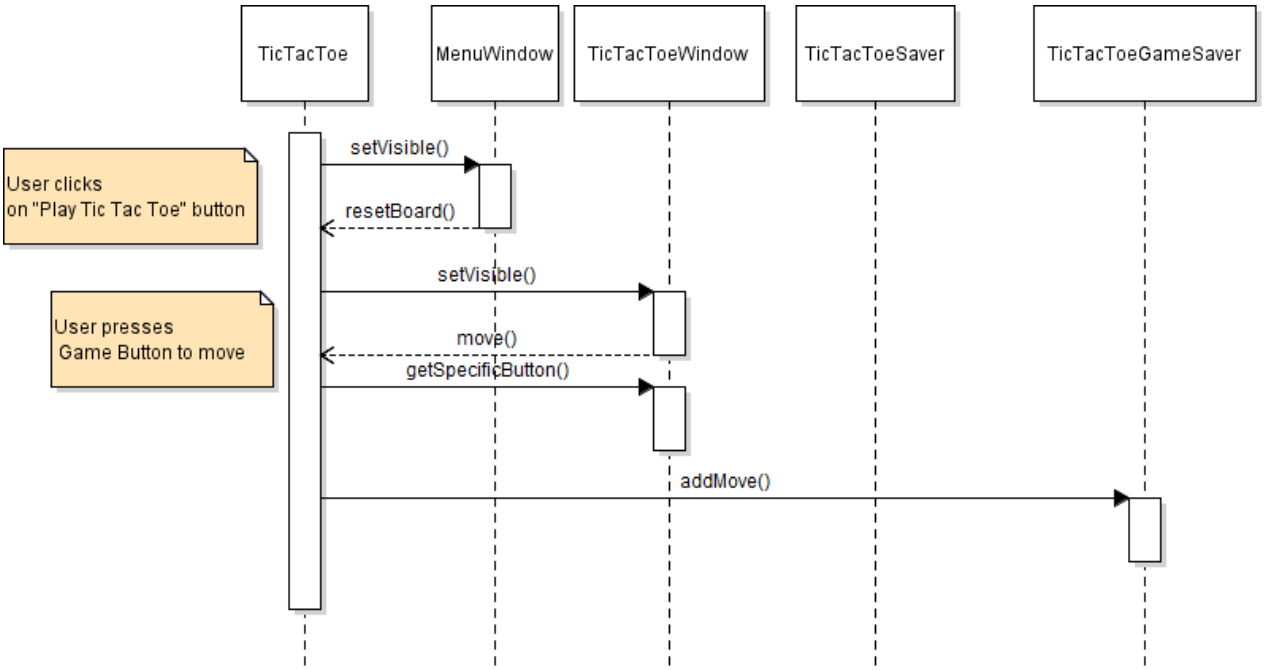


State Diagrams

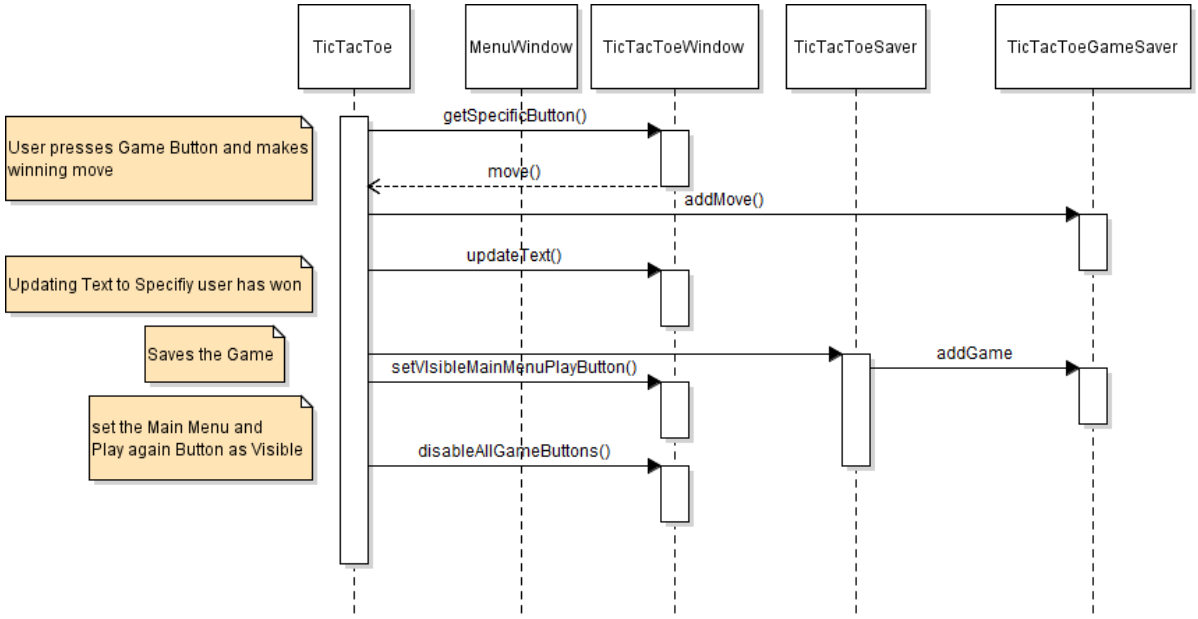


Sequence Diagrams

Making a Move in the Game



Ending A Game



Loading Past Game/ Next Move/ Back Move