

Min Ji Kim

Juan Arias

Software Development 1

May 3rd, 2017

Final Project

Abstract

In this paper, I will cover everything from why I chose to do this project to the completion of my project. I chose to code a multiplayer game that allows two users to play a game of Tic Tac Toe. This paper will also explain exactly how my code works and how that allows the game of Tic Tac Toe to be played. As someone who is learning Java for the first time, I felt like this project was a good start for me. It was a project that was not too complex but was good enough for me to spend some time sitting, thinking and actually coding the project. The main goal of this project for me is to develop a multiplayer Tic-Tac-Toe game in Java using the concepts we learned in class, such as classes, functions, OOP, and UML. This paper includes the introduction to the project, detailed system description, requirements, literature survey, user manual, and conclusion. This paper also includes how I grew from the beginning of the project to the end.

Introduction

Tic Tac Toe is a game that everyone in the world knows about. It is a game that we play while growing up and continue to play even when we are adults. As someone who is starting a new project, I wanted to code a game. A game that everyone knows and I knew the rules of is Tic Tac Toe. I always grew up playing the game, but being able to code the game by myself was

something that sparked my interest. The rules of Tic Tac Toe are rather simple. The goal of the game is to get three in a row played on a three by three game board. Two players play the game and players alternate placing X's and O's on the game board until one player has three in a row or all nine squares are filled.¹ My main goal for this project was to be able to code the game and finish it starting from nothing by utilizing the concepts I learned in class.

Detailed System Description

My final project used two classes: The TicTacToe class and the GameBoard class. The class TicTacToe contains the main method for the game. To begin the game, an empty three by three Tic Tac Toe board is displayed. Then the program continues to ask the players to enter the position in which they would like to place their X's or O's. The players will be asked to pick a position while the game is still active and the count is less than 10. The count here is used to alternate between player O and X. If $\text{count} \% 2$ equals to 0, then it would be player O's turn and if $\text{count} \% 2$ does not equal to 0, then it would be player X's turn. Every time the player is asked to pick a position and a valid position is picked, the game board will be displayed again with the corresponding player's character in the corresponding position. If position picked is invalid, players will be asked to pick again. Once every move is made, the program also checks to see if there is a winner. Program continues to ask player while the game is still active and when the count is less than 10. When the count becomes 10, the program terminates because all the spaces will be taken with no winner. Therefore, that game will be a tie.

The other class, which is the GameBoard class, is where all the rules of the games are. The GameBoard class contains two private data fields: board and active game. Board is a two-

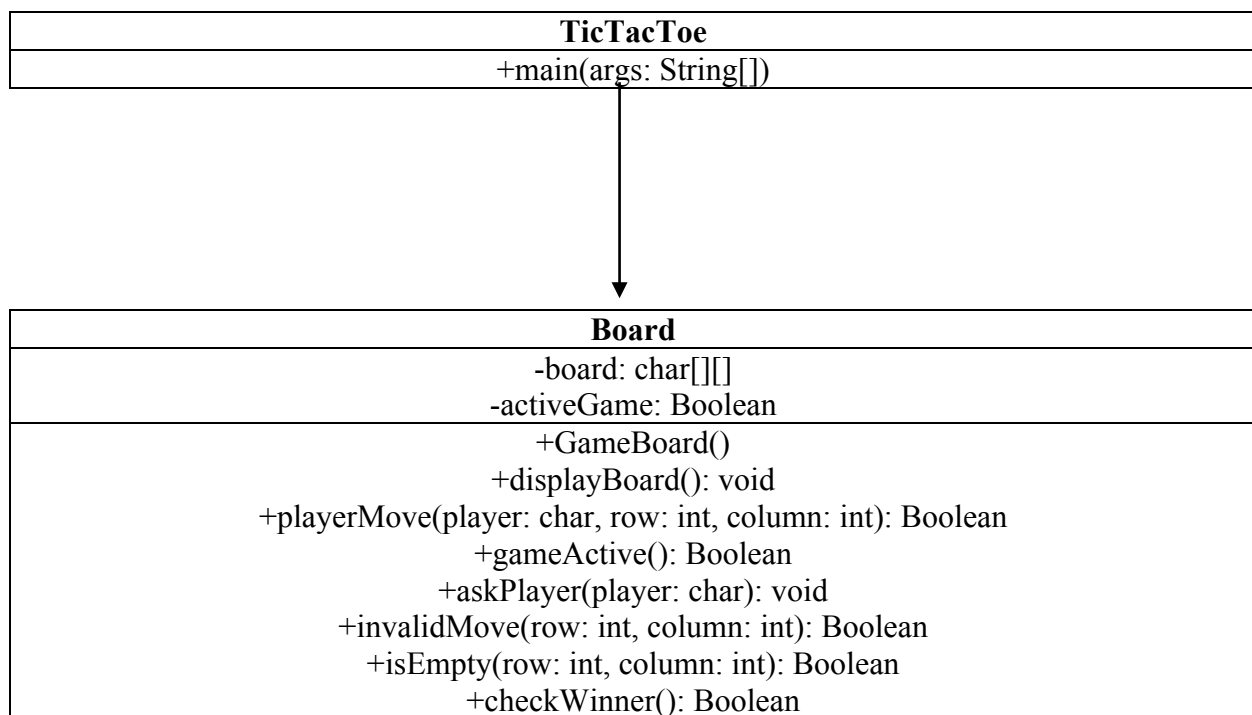
¹ <http://web.cecs.pdx.edu/~bart/cs541-fall2001/homework/tictactoe-rules.html>

dimensional array that takes in characters and the other data field is a Boolean to check whether the game is still playing. The GameBoard class has several methods. The first method GameBoard() is a constructor for the GameBoard class. In the constructor, a three by three game board is created and the game board is initialized by empty spaces. The next method is a method displayBoard() is a method that will display the board to the users. The vertical and horizontal line will only want to be printed when the rows and columns are one and zero to make it look like a Tic Tac Toe board. The next method playerMove() is a method that checks if the move that the player makes is legal. This method will return true if the move is valid and complete. Players are only allowed to enter a number within the three by three game board. Then the program checks if the position entered is available or not. If it is not available it returns false but if it is available it returns true and will make the position entered equal to the player. Then the players move is complete. The next method gameActive() is a method that checks if the game is still playing and returns the data field activeGame. The next method askPlayer() is a method that asks the player to pick a row and a column for the position to enter X or O. The Scanner is used here because the row and column values are based on what the user decides to enter. This method will keep asking the player to enter the row and column if they do not enter a valid move. Once the move is valid, it calls the playerMove() method for the other player to make the move. Here, it is important that the rows and columns each subtract one from the input because the computer reads starting from zero but normal users start from one. Whatever the user enters will be subtracted by one to match the program. The next method isValidMove() is a method that checks if the rows and columns entered are between one and three, and it also checks to see if the position the user chose is empty or not. If the position chosen matches all the requirements, then it returns true, otherwise it returns false. The next method isEmpty() checks if the chosen

position is empty or not. If the chosen position is empty, program returns true, however, if it is not empty, it will display that position is unavailable and ask the user to choose another position. Last but not least, the most important method of the program is the method that checks for the winner. There are three ways a player can win. If a player has three of the same character in a row vertically, horizontally, and diagonally, that player wins. This method checks if there are any three in a row and if it does it says who the winner is and terminates the game and once there is a winner, the game is over.

There were definitely some methods that I did not think I would need but through the help I got from the programming lab and some online resources, I was able to finish the code with all the methods required to play the game.

UML Diagram



Requirements

The requirements of the project are quite simple. This game requires two players for the game to be played. This project does not address a specific problem; however, it is a code of a game Tic Tac Toe that is played by everyone, everywhere in the world. For me, it was more important that I was able to think to myself what this game requires to be coded and work without error, than to code something that will help solve a specific problem. This project was a project to help me become more comfortable with coding and understand what is required for something to be coded.

Literature Survey

Tic Tac Toe is a well-known game so many people from all over the world has coded this game. Everyone has their own approach to this game and I had my own too. I have not really looked at other Tic Tac Toe codes but I assume that codes will contain similar methods because the game of Tic Tac Toe has relatively simple rules. Something I noticed that was different with my code and others code was that my code was a text based Java game whereas other people utilized Graphical User Interface, which has a frame that provides the main window for the GUI. From what we learned in class, I was only able to code a text based Tic Tac Toe game, however, using GUI to build a Tic Tac Toe game is definitely something that I want to do next. This will mean to learn how to use Frames and GUI on my own through online resources, however, this will make me a better programmer so it is something I want to code next. Another thing I noticed other people do was code their Tic Tac Toe game so it is played with a computer. This is also the next step I want to take to improve my coding.

User Manual

To play this text based Tic Tac Toe game in Java is does not require much work. To play this game, one must run the java program and the game will start. Once the game starts, the instructions are clearly given that allows the user to play the game until there is one winner or the game becomes a tie.

Conclusion

To conclude, this project was really important for me. Although this code may seem simple to those who have been coding for a long time, to me, coding this required me to think about all the methods that will be needed and all the rules that had to be implemented for the game to work. Before coding this, I had no idea that I would be able to code a game, but after finishing this project, it really helped me understand what is required and it also made me want to do more. While doing this project, I learned about GUI frames and that is something that I would love to learn next. Also, before starting my first project, I always thought that a program with over a hundred lines of code was a lot, but it made me realize that a hundred lines really isn't a lot. This made me become less scared of longer codes as well. I feel like although this may seem relatively simple, to me it was a great learning process. I finally learned to sit down before writing a code to think about the logic behind what I was trying to write and everything that would be required for something to work. There were times when the code would not work but sitting down and looking at the code really helped me understand my code more. I am really glad that I was able to finish this project and now I am eager to code more complex codes.

References/Bibliography

1. Liang, Y. Daniel. *Introduction to Java Programming and Data Structures*. 11th ed., Pearson Education 2018.
2. Tic-Tac-Toe Rules, Cyber Oculus Web Page, web.cecs.pdx.edu/~bart/cs541-fall2001/homework/tictactoe-rules.html
3. Programming Lab at Marist College