

Engr 102 Final Project
Lilly Woodward, Kyle Rex, Jesse Garcia

Design Document and Work Statement

Tic - Tac - Toe

Initial plan of work:

We are following a top down approach breaking the game into functions that will all work together to ensure the game runs smoothly.

Timeline:

- Meet on Saturday 12/3 to discuss the overall project and figure out who will be responsible for which part of the project.
- The majority of this work will be done before class on Monday 12/5.
- Tuesday 12/6 we will finish up the project
- Have it ready to be presented on Wednesday 12/7.

Responsible for:

- Kyle will work on the main code (valid_moves, next_move & would you like to play again) and debug everything as needed.
- Jesse will work on the main code (show_board, clear_board & play_game), the AI, and the output of the tic-tac-toe board.
- Lilly will be responsible for the Design Document, Technical Documentation and the User manual.

Functions/ Variables:

- valid_moves (input from user that is accepted for the game)
- board (the spaces meant for each "X" and "O" to go)
- play_again (if you want to play again enter "y" or "n")
- x_wins (the counter for the amount of time "X" wins)
- o_wins (the counter for the amount of time "O" wins)
- next_move (function to determine whose turn it is and to make sure the user input is valid)
- bot_move (decides next move for the easy bot)
- bot_move_med (decides next move for the medium bot)
- play_robot_easy (function that is called when you want to play the easy bot)
- play_robot_med (function called when you want to play the medium bot)
- show_board (function to print "O" or "X" in the proper space on the board)

- row_count (the row “A” “B” “C” for the user input to specify where they want to play)
- row_counter (count number of rows)
- board_print (to make the output of the board look nice)
- clear_board (function that resets the board everytime the user wants to play again)
- game_status (variable to determine whether or not the game has ended either by a win or a draw)
- winner (variable for whoever wins the game)
- play_game (a function that is called for every game of tic-tac-toe / the main function)
-

Statement of work:

- Overall we accomplished almost everything we discussed in the initial plan.
- We did run a little behind our initial schedule, but we planned extra time to ensure we had everything done.
- We encountered a few problems with the functions as we were trying to make them work properly. Once we realized what the issues were we fixed them and moved on. This project took a little longer than expected, so if we were to do it again we would definitely give ourselves more time to ensure that everything ran smoothly. We would also take some more time for the output and learn how to use AI for a nicer presentation. Although we did not have time for this, we did make our output as clean and neat as possible.
- Kyle
 - valid_moves (to ensure the move enter was valid and if it was not to ask user for a new input)
 - next_move (function to go back and forth between players and make sure moves were valid)
 - play_again (set up the play again feature if users wanted to play more than once)
 - Helped debug and clean up the code
- Jesse
 - show_board (function to produce the output of the board after a player made a move)
 - clear_board (function that resets the board everytime the user wants to play again)
 - play_game (a function that is called for every game of tic-tac-toe / the main function)
 - Made the counter for who has won each game in a row
- Lilly

- Made the user manual
 - Made the technical document
 - Made the design document and statement of work
 - Helped debug and clean up the code
- There were a few things that we could have done a little better to make this project go smoother. One is at the beginning of the project we should have laid out the overall structure of the code so there would have been less confusion while making it. If we would have set a better expectation for each other, we think we could have gotten a lot more done in a shorter amount of time. Secondly, we think if we would have made specific due dates for ourselves to finish each part of the project, it would have been less overwhelming. We also would have kept us with what everyone was doing and when they were going to be done with it easier. Overall, our group worked well together and communicated to each other to finish the project.

Creation Timeline:

- Created board
- Function to update board with moves of player's decisions
- Function to detect the win of either X or O, or draw
- Created function to start another game if player decides
- Clear_board function to reset valid moves and empty the board for replay
- Implemented a function that calls the functions to show board and ask for a move called play_game (essentially main function for player vs player)
- Base game created now to implement AI.
- Created two new functions for the Easy AI in which one chose the move (random valid move) and the other actually putting the move on the board
- Involved some debugging and a good amount of trial and error implementing this function into existing functions
- Created two new functions for a Medium AI in which one chose the move (Try to block a players winning chances (sometimes)) and go for the win if it can (sometimes), the other just put the move on the board.
- Same bunch of debugging not only implementing it to the base game but also figuring out how to calculate the best moves