**Meeting The Brief**

(Video Goes Here)

**Investigation and Plan**

I started my investigation by looking at many different board and card games. I used search engines such as DuckDuckGo and Google to find info on many board and card games. I looked at Snakes and Ladders, but the game is only luck based and there is no way for players to improve and get better at the game. I looked at Solitaire, Uno and Chess and these games would work for this project, but I think that they are too complicated for this project and take up too much time to program. I settled on doing Tic Tac Toe because it has the ability for players to get better at it and the game has a bit of probability to it too. I have looked at the rules and other digital versions of Tic Tac Toe and how they work (see references). I decided to not do graphics and have the game be rendered in the command line as I don't have the time to learn and make graphics for the game. My project will be written in Python as it is the programming language I know best, it will be rendered in the command line and it will be mostly written in a functional programming way to make the program as easy to code and modify as possible. My objectives are to make a working project that hits the below requirements:

**Meeting Basic Requirements**

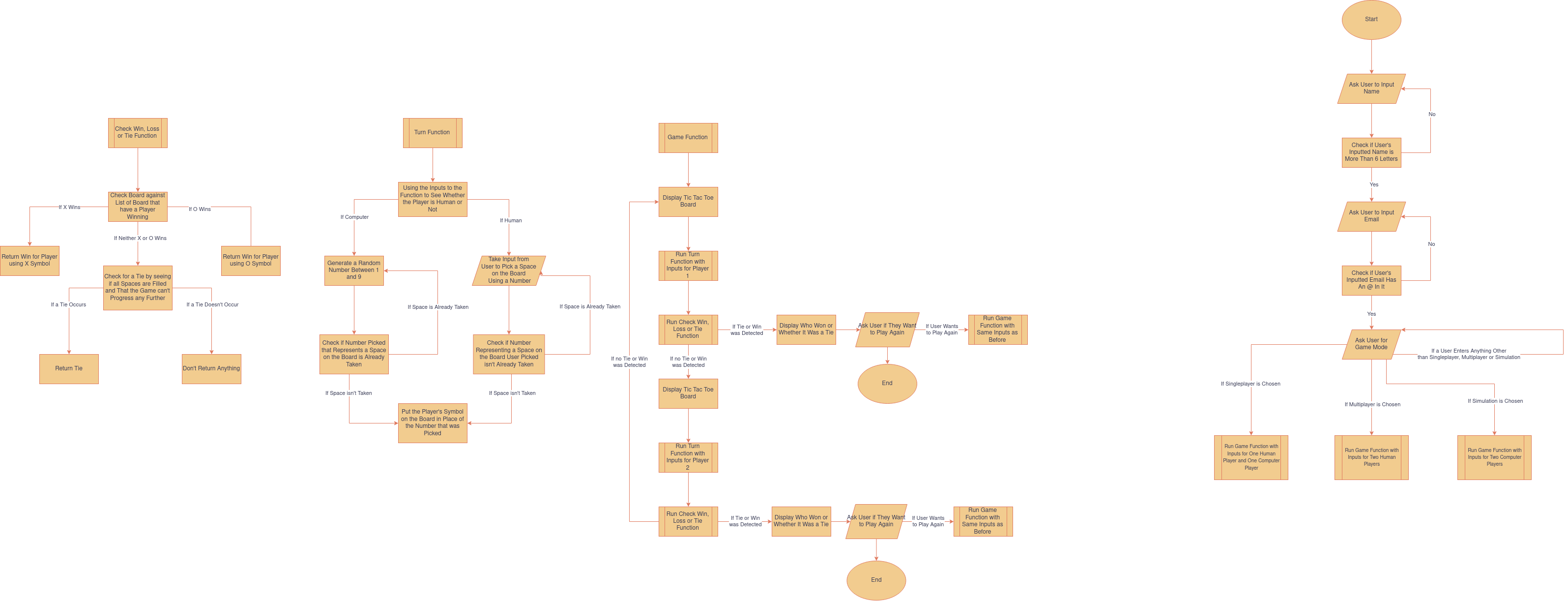
* Tic Tac Toe is a game that isn’t too complicated, and I can model and create in python in a moderate amount of time
* Tic Tac Toe can have many inputs for the game itself and the game can start by taking name and email from the user as inputs
* Tic Tac Toe could be programmed for the player to be able to play the game on their own against the computer, have two players play against each other and I can program it to have two computers play against each other

**Meeting Advanced Requirements**

* All of each player’s Tic Tac Toe moves and which player wins I will be able to record into a csv file
* The data generated from Tic Tac Toe is stored in a csv file and can be used for performing statistical analysis using algorithms to get info like the frequency, mode, mean and median of a player’s moves.
* The above data can be used to prove or disprove a hypothesis like “If player 1 puts their first move in the middle of the board, they are more like to win”

**Design**

**A clear detailed description of how the project will be developed.**

First, I thought about all my requirements and how I will meet them. To take 3 inputs, I will have the program take three inputs: the user’s name, the user’s email and the game mode the user wants to play in. Next, I decided to make the main game in the program as a function with inputs that will decide which players are human or the computer. If a player is human, it will ask the user for an input to pick the player’s choice for the board. If the player is the computer, it will generate a random number between 1 and 9 as the player’s choice for the board. If the player's choice on the board is already occupied, a human player will be asked to input a choice for an empty spot on the board **A description of how abstraction, it will just generate a nother numbernd modelling will be applied.**

To Do

* Use words: agile programming, decomposition and abstraction
* Had to make game easier, come up with something to make it seem like I made it somewhat easier (Can’t click square to input move, so instead you input using numbers)

**Implementation**

\_

To Do

* Describe making the program
* Describe a problem/error I ran into and how I solved the problem/error: using try loop to clean csv data and not error out when positions don’t work, and it can’t add it to the list

**Testing**

An overview of the testing carried out.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Action | Input(s) | Expected Output(s) | Actual Output(s) | Test Result | Test Data Used | Comments |
| Accept and validate username | Any name that is 6 or more characters long | Accepts a valid name and moves on | Accepted a variety of names containing various characters that were 6 or more characters long and moved on | Pass | “John,” “Mary”, “(^\*$”, “c00lj0e78”, “أَمِيْن” and “明美” | It also accepted names with symbols, numbers and even non-English characters |
| Accept and validate email | Any input that contains the “@” character | Accepts a valid email and moves on | Accepted a variety of strings, if the string contained an “@” and moved on | Pass | “Jim@yahoo.com”, “coolbeans@gmail.com”, “jim@yahoo” and “beans\_hotmail.com” | It also accepted emails that weren’t real or had a proper domain e.g., it accepted jim@yahoo, while it should be jim@yahoo.com |
| Accept and validate game mode | The numbers 1 to 5 | Launches game mode that the number the user inputted corresponds with | Accepts the numbers 1 to 5 and launches the game mode associated | Pass | 1, 2, 3, 4, 5, 6, 0, “one”, “two”, “zero”, “four”, “three”, “five”, “Singleplayer”, “singleplayer” and “multiplayer” | It didn’t accept the words “one”, “two”, “three”, “four” or “five” and it didn’t accept the names of the game modes |
| Main game | Player’s move | After each player move it shows board or if a player won | If a player has won, it displays that that player has won and if no player has own it shows board and then accepts the next player’s move and this repeats until someone wins | Pass | 3, 2, 5, 4, 1, 9, 0, “nine”, “ten”, “Four” and “Foot” | It accepted any number from 1 to 9, but not 0 as there is no square labeled 0. It also didn’t accept the strings “nine”, “ten” or “Four” |

**Evaluation**

\_

**References**

1. [DuckDuckGo](https://duckduckgo.com/) - Search Engine to find out information, references and resources
2. [Google](https://www.google.com) - Search Engine to find out information, references and resources

**Games I Considered Doing**

1. [Snakes and Ladders](https://en.wikipedia.org/wiki/Snakes_and_ladders) – This game is only luck based and there is no way for the players to get better
2. [Solitaire (Patience)](https://en.wikipedia.org/wiki/Patience_(game)) – This game I found to be too complicated and/or time consuming to program for this project
3. [Chess](https://en.wikipedia.org/wiki/Chess) - This game I found to be too complicated and/or time consuming to program for this project
4. [Uno](https://en.wikipedia.org/wiki/Uno_(card_game)) – This game I found to be too complicated and/or time consuming to program for this project

**Game I Picked (Tic Tac Toe)**

1. [Rule of Tic Tac Toe and Online Version of Tic Tac Toe](https://www.exploratorium.edu/brain_explorer/tictactoe.html) - Rules on how Tic Tac Toe works and an online version to get used to how the game works
2. [Online Tic Tac Toe Game on Google](https://www.google.com/search?q=tic+tac+toe) - Another online version to get used to how the game works

**Tools Used to Make Artefact**

1. [Visual Studio Code](https://code.visualstudio.com/) - The IDE I used to program the artefact
2. [Visual Studio Code Python Extension Pack](https://marketplace.visualstudio.com/items?itemName=donjayamanne.python-extension-pack) - Extension pack I used to help with programming by adding colouring and debugging for python to Visual Studio Code

**Links Used to Solve Problems**

1. <https://www.w3schools.com/python/python_functions.asp>
2. <https://stackoverflow.com/questions/14594522/how-to-break-out-of-while-loop-in-python>
3. <https://duckduckgo.com/?t=ffab&q=tic+tac+toe+ascii&atb=v324-1&iax=images&ia=images>
4. <https://www.geeksforgeeks.org/declare-an-empty-list-in-python/>
5. <https://www.w3schools.com/python/python_lists.asp>
6. <https://stackoverflow.com/questions/9573244/how-to-check-if-the-string-is-empty>
7. <https://stackoverflow.com/questions/62379898/greater-than-or-equal-vs-equal-or-greater-than-in-python#62379949>
8. <https://www.tutorialspoint.com/python/python_command_line_arguments.htm>
9. <https://www.geeksforgeeks.org/find-length-of-a-string-in-python-4-ways/>
10. <https://stackoverflow.com/questions/53477929/check-if-string-contains-specific-character#53478137>
11. <https://stackoverflow.com/questions/5423381/checking-if-sys-argvx-is-defined>
12. <https://www.w3schools.com/python/python_for_loops.asp>
13. <https://pythonguides.com/python-check-if-the-variable-is-an-integer/>
14. <https://stackoverflow.com/questions/64097627/check-if-string-is-in-a-list>
15. <https://realpython.com/convert-python-string-to-int/>
16. <https://www.geeksforgeeks.org/convert-string-to-integer-in-python/>
17. <https://www.w3schools.com/python/ref_string_isdigit.asp>
18. <https://stackoverflow.com/questions/306400/how-can-i-randomly-select-an-item-from-a-list#306417>
19. <https://www.geeksforgeeks.org/python-replace-substring-in-list-of-strings/>
20. <https://www.pythoncentral.io/how-to-generate-a-random-number-in-python/>
21. <https://stackoverflow.com/questions/21388541/how-do-you-check-in-python-whether-a-string-contains-only-numbers#21388567>
22. <https://www.geeksforgeeks.org/python-check-if-two-lists-are-identical/>
23. <https://www.w3schools.com/python/python_booleans.asp>
24. <https://stackoverflow.com/questions/28254807/how-to-loop-back-to-the-beginning-of-a-programme-python#28254828>
25. <https://www.geeksforgeeks.org/python-check-if-a-list-exists-in-given-list-of-lists/>
26. <https://www.pythonforbeginners.com/basics/list-of-lists-in-python>
27. <https://duckduckgo.com/?t=ffab&q=tic+tac+toe+all+possible+wins&atb=v324-1&iax=images&ia=images>
28. <https://pythonexamples.org/python-if-and/>
29. <https://www.w3schools.com/python/ref_keyword_continue.asp>
30. <https://duckduckgo.com/?t=ffab&q=all+tie+possiblitys+in+tic+tac+toe&atb=v324-1&iax=images&ia=images&iai=https%3A%2F%2Fi.imgur.com%2FLhxdJ0L.png>
31. <https://www.w3schools.com/python/python_file_write.asp>
32. <https://www.w3schools.com/python/ref_list_append.asp>
33. <https://stackoverflow.com/questions/5618878/how-to-convert-list-to-string#5618893>
34. <https://docs.python.org/3/library/statistics.html>
35. <https://www.geeksforgeeks.org/python-read-csv-columns-into-list>
36. <https://stackoverflow.com/questions/32441633/how-to-read-specific-column-index-from-csv-using-pandas>
37. <https://stackoverflow.com/questions/82831/how-do-i-check-whether-a-file-exists-without-exceptions#82852>
38. <https://duckduckgo.com/?q=column+vs+row&t=newext&atb=v339-1&iax=images&ia=images>
39. <https://docs.python.org/3/library/statistics.html>
40. <https://stackoverflow.com/questions/42788930/how-to-take-only-first-four-elements-of-every-strings-from-a-list>
41. <https://stackoverflow.com/questions/11786157/if-list-index-exists-do-x>
42. <https://realpython.com/python-rounding/>
43. <https://stackoverflow.com/questions/53513/how-do-i-check-if-a-list-is-empty>
44. <https://stackoverflow.com/questions/2161752/how-to-count-the-frequency-of-the-elements-in-an-unordered-list#2162045>
45. <https://www.geeksforgeeks.org/python-ways-to-remove-n-characters-from-start-of-given-string/>