**Knowledge and Tools:**

The game is developed using basic algorithmic knowledge and rules of the game as per defined in the instructions file. The analytical problem solving approach is used to fulfill the requirements of the game. Development is done using Python programming language and Pycharm IDLE.

**Data Structures:**

The most important data structure used for this game includes Dictionaries, tuples, Lists. Out of three of these the most extensively used is List Data Structure as it’s the most common and easy to use out of all the data structures in Python.

**External Libraries:**

Three external libraries are used in this development including Itertools, to get cycle for the repetitions of turns between players, Collections for using a named tuple and Strings to get the asci for Uppercase alphabets.

**Implementation:**

The Game basically consists of 4 python files and one text file (which will be created after you complete one game play). The python files are divided to simplify the code study and for better understanding. The main file from where the execution begins is “battelships.py”. The file consists of the main function and execution class named Game. The main function starts and asks the user to choose between Player vs Player or Player Vs Computer Game Play.

Option 1:

If the user choses option 1 he is sent to class Game. This class handles all the execution of Player Vs Player Gameplay. It starts with displaying a welcome note, previous scores if exists and then the user is asked to give inputs. The Boards are generated and then one by one users are asked to place their ships on the boards. After placing the boards players guess locations and play the game. Once the game is finished the user is sent back to main function he will again give his/her choice to play either Player Vs Player or Player Vs Computer. This Class or Player Vs Player game play other two files also. File named “board.py” is used to make boards and check and place and generate the ships. It also converts the locations to computer readable location. The other file used by this class is “const.py”. This is basically the declaration of all the constants being used by the other two files.

Option 2:

If the user choses option 2 he is sent to the forth and last file named “pve.py”. This is basically a complete game in itself as it doesn’t take any information from other files. The user is welcomed and asked to place all the five ships and then the board is displayed and user start playing against the computer. This file also handles the placing of computer ships and it’s gameplay which is explained in next section of this document.

PC gameplay "Intelligence" other extra implementations:

In Player vs Player Selection of user, the user is playing with computer. The computer places it’s ships itself and attacks the user ships brilliantly. Here is how ?

The computer program is written and all the possibilities involved are added in this section. The computer firstly starts placing the ships. It guess a random x-axis and y-axis between the given range of board and then checks if that locations exist in the board. If yes then it checks if the place is already occupied by some ship or not. And then finally it places the ship. The computer can place 5 ships just like user and also computer randomly select between horizontal or vertical position before placing the ships. Once the ships are and user starts it’s first attack now the computer is called to play it’s turn. Now computer generate a random guess within the board range and hit that place. This is how the computer play smartly against the user. Isn’t it cool ?

**Sources:**

**https://docs.python.org/3/tutorial/inputoutput.html**

**User Manual:**

1. The Game will ask you to choose between two options (Player Vs Player and Player Vs Computer). You have to choose between the two options. Enter either 1 or 2.
2. While playing player vs player, Enter Your names first and then place the ships by giving the starting position as A1 or D6 etc. After that you must confirm the placement of ship by entering a small or capital “ C ”.
3. When you are asked to choose the location to hit then enter any location like A1, F5, J3 etc. just like before.
4. If you select option 2 from main menu then start you game by entering the ships.
5. Firstly you have to enter the Horizontal or vertical position of the ship by entering 1 or 2.
6. After that you have to give the start position coordinates of the ship as x-axis and y-axis.
7. Similarly when you have to hit a ship you will chose the x-axis and y-axis position.
8. At the end of each game you will be sent back to the main menu to ask if you want to play again or not.