Software Requirement Specification – Tic Tac Toe Game

1. Introduction
   1. Purpose: The tic-tac-toe game demonstrate skills in using Python for solving complex problems by integrating programming techniques in one program consisting of various parts.
   2. Intended Audience: the SRS document is to be accessible by the General public through GitHub
   3. Intended Use: the source code can used per the license for this project. The license is available in the project folder
   4. Scope: the goal of this project is developing a console software application to play tic-tac-toe with the end-user
2. Overall Description
   1. User Needs
      1. The computer should play the game using the ‘X’ symbol
      2. The user should play the game using the ‘O’ symbol
      3. The first move belongs to the computer
      4. The computer picks a first move in a pseudorandom approach
      5. All squares are numbered row by row starting with ‘1’ and ending with ‘9’
      6. The user inputs their move by entering the number of the square they choose - the number must be valid. It must be an integer, it must be greater than ‘0’ and less than ‘10’, and it cannot point to a field which is already occupied.
      7. The program checks if the game is over, and decided to continue the game, end the game with one side wining, or end the game because there is a tie between the computer and the player
   2. Dependencies
      1. The user needs to have a Python interpreter installed on the machine to execute the program
   3. Activity diagram representing one game session

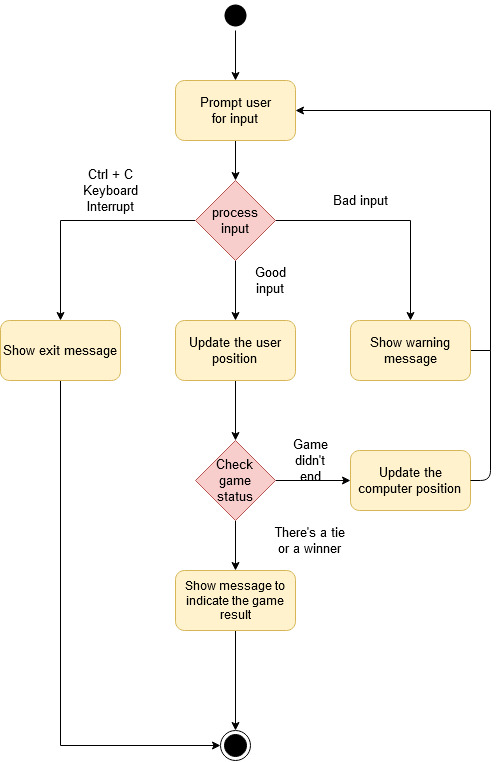


Figure 1

1. System Features and Requirements
   1. Functional Requirements
      1. If the user inputs an integer outside the minimum and the maximum allowed row number, then the program must warn the user and give the user another chance to give an input
      2. If the user inputs a value for an already occupied field, the program must warn the user and give the user another chance to give an input
      3. If the user inputs a non-integer value, the program must warn the user and give the user another chance to give an input
      4. If the user wants inputs “Ctrl + C” to exit the program, the program must exit gracefully by not giving an error
   2. External Interface Requirements
      1. The padding for the board game must be symmetrical across the x-axis and symmetrical across the y-axis
   3. Nonfunctional requirements
      1. The code for the program must meet PEP8 standards with at least 95 % rating