Tic-Tac-Toe using Python

Submitted By

Name	ID
Faria Sultana	C201203
Faiza Binta Ali	C201213
Khadizatul Kubra	C201225
Farida Nusrat	C201242

Course Code: CSE-3638 + CSE-3638

Course Title: Software Engineering & Software Development Lab

Dept. of Computer Science & Engineering.

International Islamic University Chittagong.

Table of Content

Project Name: Tic-Tac-Toe using Python

Project Description: This game is very popular and is fairly simple by itself. It is actually a two-player game. In this game, there is a board with n x n squares. In our game, it is 3 x 3 squares. The goal of Tic-Tac-Toe is to be one of the players to get three same symbols in a row - horizontally, vertically, or diagonally - on a 3 x 3 grid.

A player can choose between two symbols with his opponent, usual games use "X" and "O". If first player choose "X" then the second player have to play with "O" and vice versa.

A player marks any of the 3x3 squares with his symbol (may be "X" or "O") and his aim is to create a straight line horizontally or vertically or diagonally with two intentions:

a) Create a straight line before his opponent to win the game.

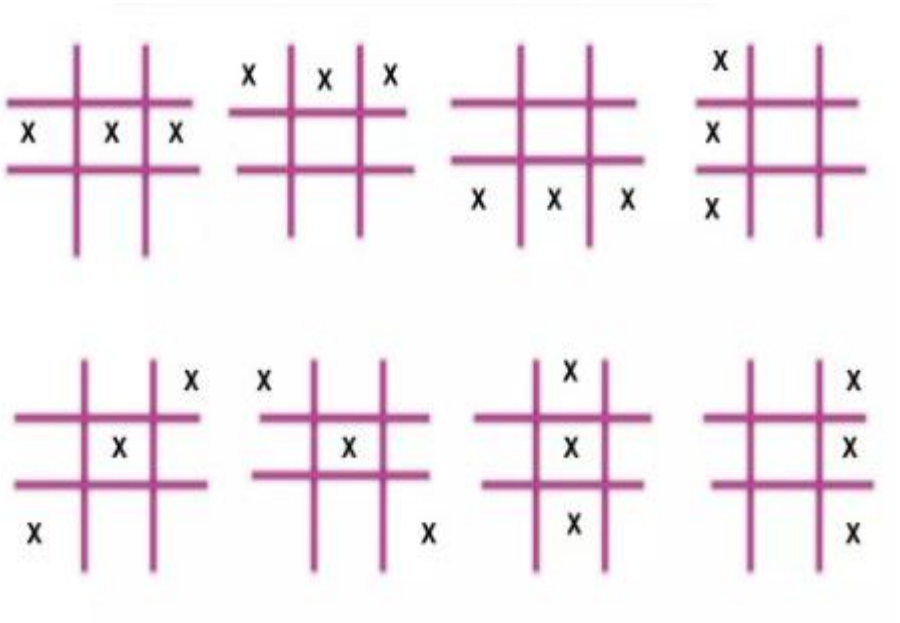
b) Restrict his opponent from creating a straight line first.

In case logically no one can create a straight line with his own symbol, the game results a tie.

Hence there are only three possible results - a player wins, his opponent (human or computer) wins or it's a tie.

1	2	3
4	5	6
7	8	9

If any player is able to draw three Xs or three Os in the following combinations then that player wins. The combinations are:



Requirements:

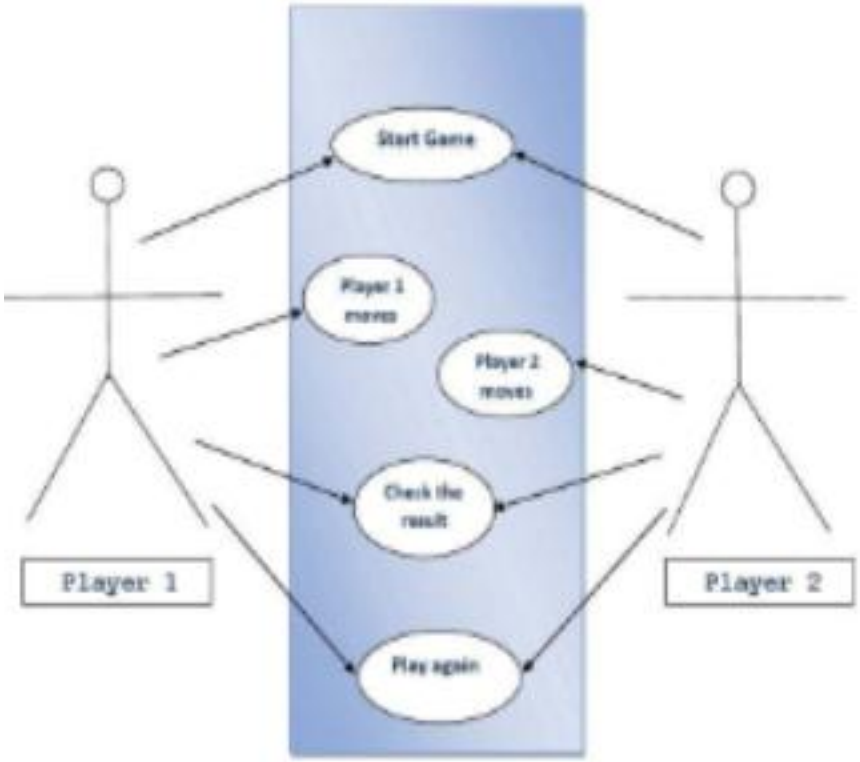
Hardware requirement [minimum requirement] :

- ✓ Minimum RAM:- 1GB
- ✓ Minimum Hard Disk: -128GB
- ✓ Processor: - Intel Pentium 4(1.50 GHz) or above

Software requirement [minimum requirement] :

- ✓ Operating System: - Support for WINDOWS users
- ✓ Back End: - Python 3.6.0 Interpreter
- ✓ Front End Language: - Python3
- ✓ Front Design: - Tk interface

Use case Diagram:



Design:

Input design: Input design is part of overall system design that requires special attention designing input data is to make the data entered easy and free from errors. The input forms are designed using the controls available in PYTHON INTERPRETER. Validation is made for each and every event that is happened. Help (how to play the game) information is provided for the users during when the players feels difficult.

Input design is the process of converting the user originated inputs to a computer-based format. A system user interacting through a workstation must be able to tell the system whether to accept the input to produce reports. The collection of input data is considered to be most expensive part of the system design. Since the input has

to be planned in such a manner so as to get relevant information, extreme care is taken to obtain pertinent information.

Output design: The output design this application "TIC TAC TOE" generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

The output is designed in such a way that it is attractive, convenient and Informative. Board is designed with various features, which make the console output more pleasing. As the outputs are the most important sources of Information to the users, better design should improve the system's relationships with us and also will help in decision making. Form design. elaborates the way output is presented and the layout available for capturing information.

One of the most important factors of the system is the output it produces. This system refers to the results and information generated. Basically, the output from a computer system is used to communicate the result of processing to the user.

Test plans: The test approach is divided into three main phases: Module testing, integration testing and system testing. In addition, the system testing includes two sub-phases: functional and usability testing. These planned tests are explained briefly below.

(a) **Module testing** will perform during coding by using debug messages to check that the written code produces wanted results. An important requirement is that the code will compile with zero bugs.

(b) **Integration testing** will perform after finish module testing in order to validate if each module can work fine with each other. Integration Test proves that system works as integrated unit when all the fixes are complete.

(c) **System testing** includes two phases: functional testing and usability testing. These will perform after the product reaches its final version. During the functional test phase, the tester will test if the product meets the game requirements. The tester tests the requirements using the use cases listed below in Test Cases section. The usability test will perform to understand how easy it is to learn to play the game. Any person out of the team members will perform this test by playing the game.

Limitation:

1. If any button of the 3×3 grid is filled by a player then that button cannot be filled again by that player or another player in a game.

2. If one player fills a button in the grid that player cannot remove it until that player restarts the game.

Conclusion: We have successfully developed the Tic-Tac-Toe game project in python. We use tinker module for rendering graphics on a display window. We learn how to create buttons and config text on buttons and also how to use lambda functions to send specific values to callback functions.

In this way we successfully made a Tic-Tac-Toe game python project.