

Multiplayer Tic Tac Toe with AI Support

Project Report

Group Member:

Sharjeel Safder 22k4210

Tulaib Tauseef 22k 4437

Introduction

This project is a Streamlit application of a better version of Tic Tac Toe game, built with Python. It can support between 2 and 5 players – and it features AI-controlled players, so that human-human and human-AI action is possible. The game is highly configurable with symbols and variant sizes of grid, as well as consecutive symbols winning rules.

Objectives

- Construct a multiplayer game which can be played online using the browser.
- Support between 2 to 5 players.
- Contain CI controlled opponents (basic random strategy).
- Permit modification of player symbols, size of the board.
- Show the game board in an interactive manner with real feedback.

Key Features

1. Customizable Game Setup

Choose number of players (2–5).

Modify custom symbols according to each player.

Switch to Human and then to AI for every player.

Adjust grid size (5×5 to 9×9).

Pick win condition (3 – 5 symbols in a row).

2. Interactive Game Board

Dynamically generated into this interface by using Streamlit layout components.

Players will move by clicking on empty cells.

AI users make automatic moves through the random selection.

3. AI Support

AI selects random available cells.

With a view to ease, but could be extended to wiser strategies.

4. Winner Detection

Matches rows, columns and diagonals for sequence.

Immediately tells the winner.

5. Session Management

Takes advantage of `st.session_state` to store game state while rerunning the game.

Keeps board status, player turns and winner's info.

Technical Stack

Frontend/Backend: Python + Streamlit

State Management: Streamlit Session State

Libraries: NumPy (used optionally), random

Code Structure

Game Setup

Controlled via the sidebar.

Sets or resets the game depending with the entry from the user.

Winner Checking

Function `check_winner()` reads for sequences in rows, columns and diagonals.

Game flexibility is enhanced by the fact that win length can be configured.

AI Logic

`ai_make_move()` function chooses randomly an empty cell and plays.

Makes sure that AI players get decisions made quickly and fairly.

Game Loop

Displays current turn.

Paints the game board in a row by row fashion.

Solves both AI as well as user interaction.

Finishes with a victory message or lets reset.

Possible Improvements

Apply wiser AI (Minimax, heuristic-based);

Insert into game history and undo capabilities.

Support dark mode and the feature of mobile responsiveness.

Add player statistics tracking.

Deployed online on Streamlit cloud or Heroku.

Conclusion

This project is live evidence of how easy it is to create interactive web applications without much overhead using Streamlit. With AI integration and multiplayer, it makes an entertaining and easy to use experience, which can be further expanded into higher end games.