TIC-TAC-TOE

NAME- KHUSHI CHOUDHARY

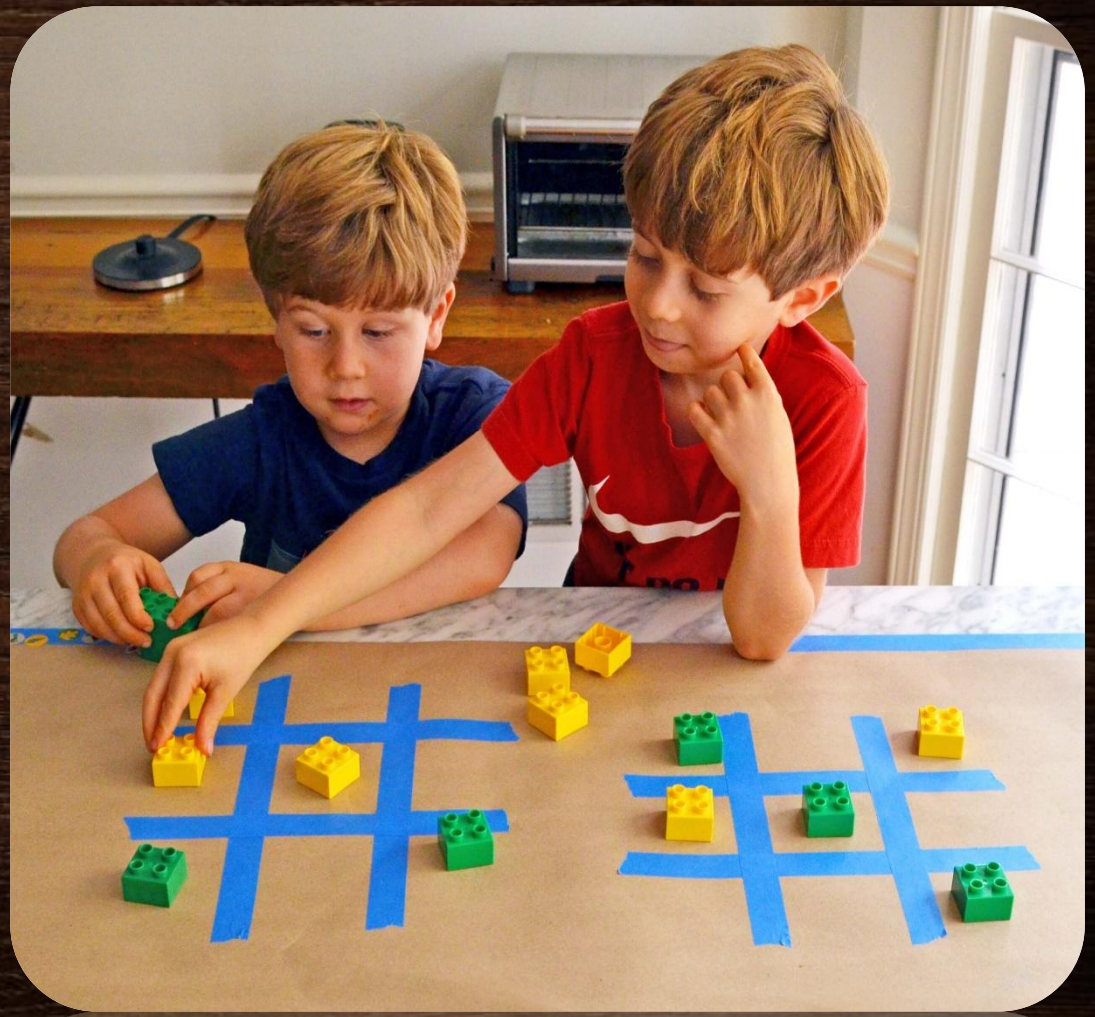
MICRO IT PROJECT

Introduction

Tic-Tac-Toe is a timeless and simple strategy game that has been used for years as a beginner-level programming project to introduce logic building, user interface design, and event handling. During my internship, I was tasked with creating a fully functional version of Tic-Tac-Toe that supported interactive play, intuitive design, and efficient logic. This project helped me strengthen my skills in programming, game development, and software design principles.

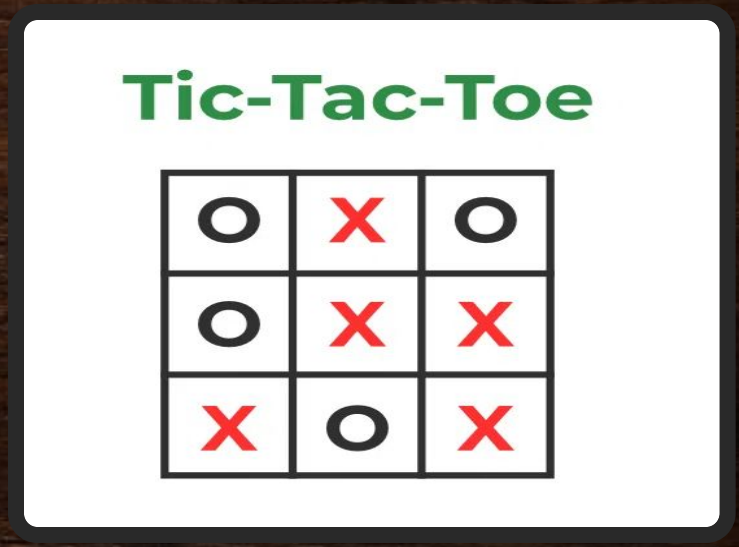
Objective of the Project

The primary goal of this project was to : Understand the software development life cycle by creating a simple yet complete application. Implement core programming concepts like conditionals, loops, functions, and arrays. Learn basic game development and UI/UX concepts. Provide an interactive platform that allows two users to play the game on a single device.



Tools and Technologies Used

The development of the Tic-Tac-Toe game involved the use of the following technologies.

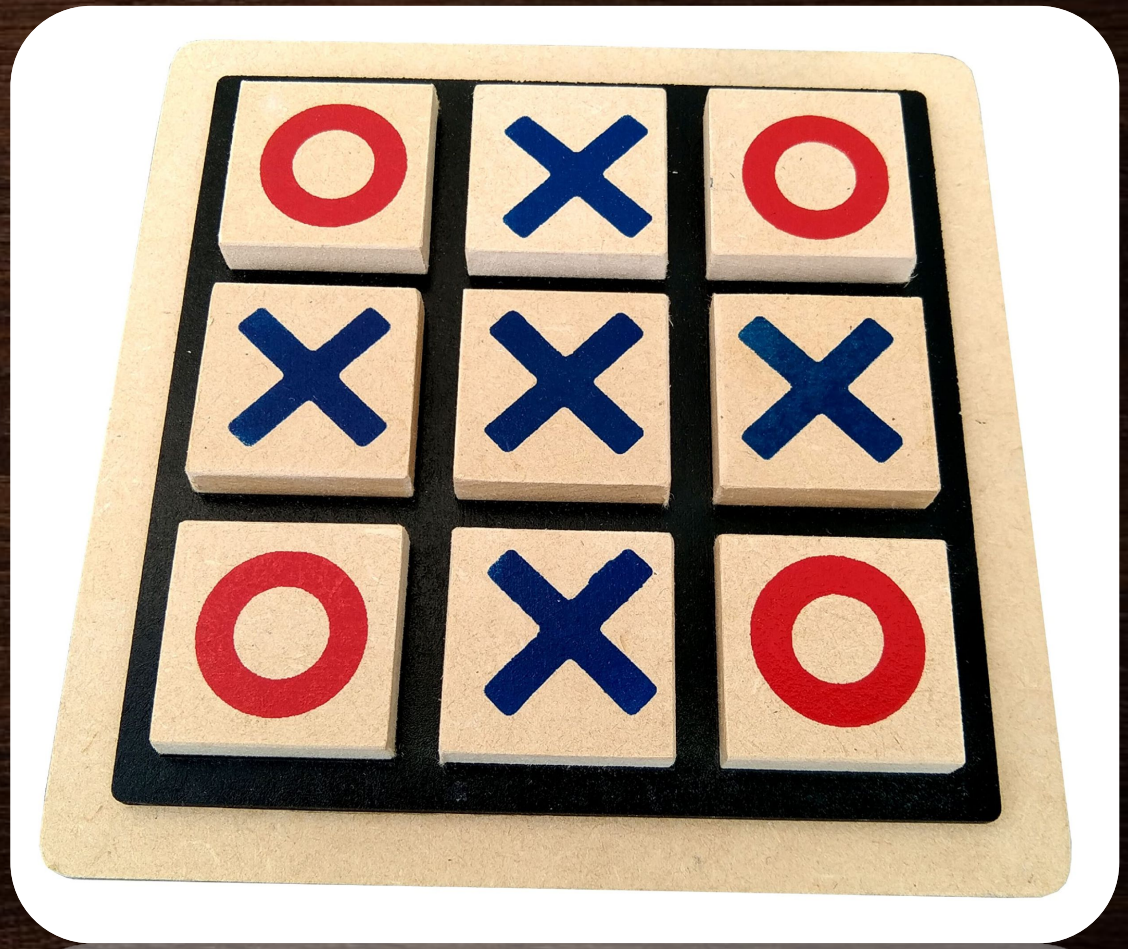


NOTE : YOU CAN REPLACE THE TOOLS ABOVE BASED ON WHAT YOU ACTUALLY USED.

SYSTEM DESIGN AND ARCHITECTURE

Game Flow:

- The game starts with an empty 3x3 grid.
- Players alternate turns, marking either X or O in the cells.
- After each move, the system checks for a win or draw.
- If a player wins or the grid is filled [draw], the game ends with a message.

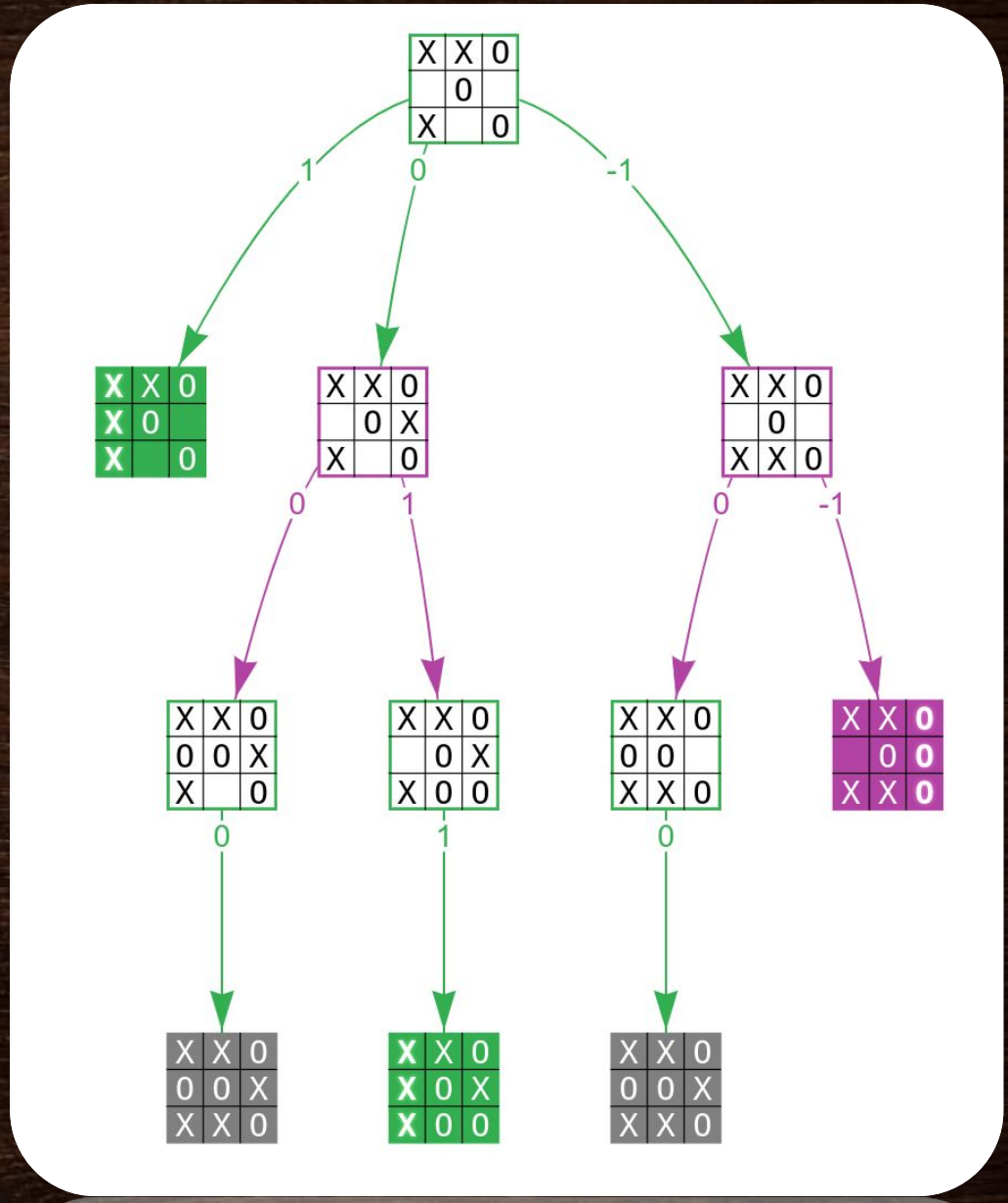


User Interface :

- Clean and minimalist design
- Start/Restart button
- Player turn display
- Result pop-up or message after win/draw

START

RESTART



Implementation Details

1. LOGIC :

- A 2D array [or a single list] is used to track the game state.
- A function checks all rows, columns, and diagonals after each move.
- A move is considered valid only if the cell is empty.

2. KEY FUNCTIONS :

- check_winner[]: Verifies if a player has won.
- check_draw[]: Checks if the game is a draw.
- reset_game[]: Resets the grid for a new match.

3. CODE STRUCTURE :

- Main File: Handles UI and game loop
- Logic Module: Contains backend game logic
- Assets: [if applicable] stores visuals/sounds

Challenges Faced

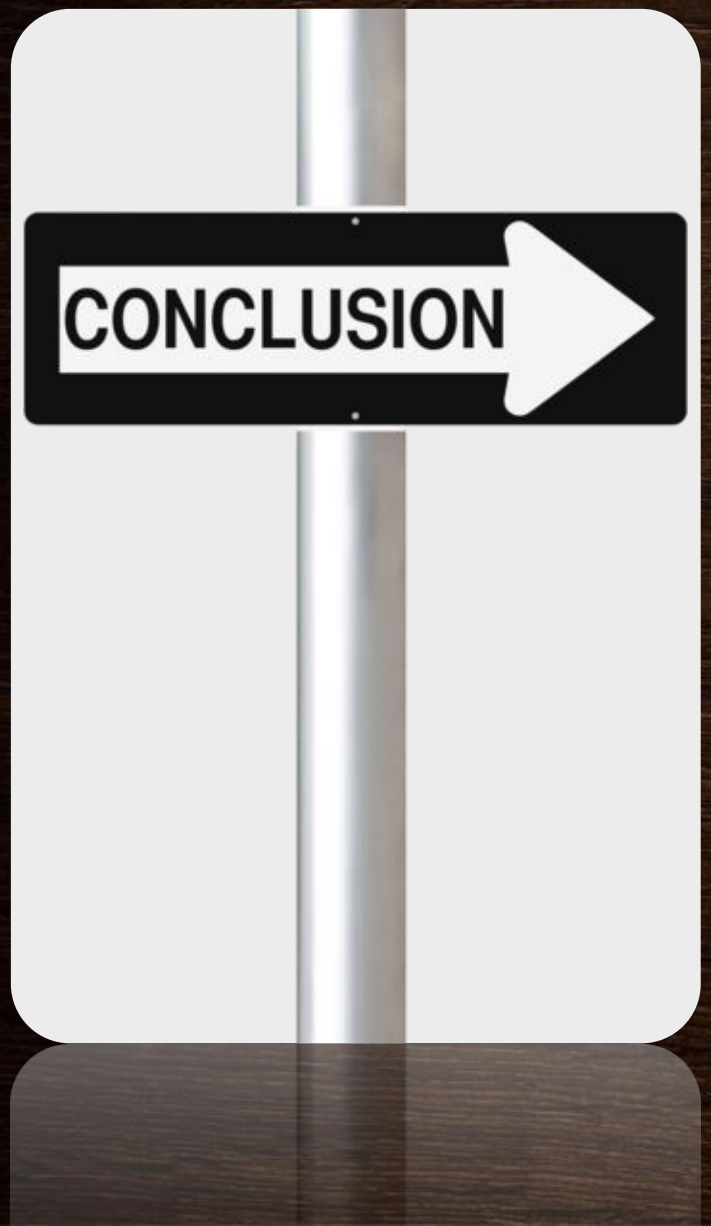
- Managing UI responsiveness for different devices.
- Handling edge cases (e.g., double-clicking same cell).
- Ensuring real-time updates of player turns.
- Debugging win-check logic for all conditions.

LEARNING OUTCOMES

This internship project taught me:

- How to structure and manage a small-to-medium scale software project.
- The importance of clean code and modular programming.
- Hands-on experience in creating user-friendly interfaces.
- Basics of software testing and debugging.
- Effective use of version control systems.

Conclusion



This Tic-Tac-Toe project helped me develop both technical and problem-solving skills.

Despite being a relatively simple game, the development process was rich in learning and practice.

From UI design to backend logic and from error handling to version control, the project gave me a real-world feel of software development.

Future Scope

Some possible enhancements include:

- Adding AI for single-player mode
- Keeping track of scores over multiple rounds
- Making the game responsive for mobile devices
- Adding animations or sound effects for better user experience