University of Science and Technology of Southern Philippines

Tic-Tac-Toe

Peter Christian Almoroto
Ulysses Joi Almoroto
Jasper Amper
Justine Macayanga
Adrian Manigos
Arvell Galarpot

An Object Oriented Programming Project

Department of Information Technology

College of Engineering and Technology

University of Science and Technology of Southern Philippines – Claveria Campus

Claveria Misamis Oriental, 9004 Philippines

December 9. 2024

I. Background of the Study

Tic Tac Toe, a classic two-player strategy game, serves as an engaging platform for logical reasoning and decision-making. This project is designed to replicate the game using object-oriented programming (OOP) principles. By leveraging OOP concepts such as classes, objects, encapsulation, and inheritance, the development process emphasizes modularity, reusability, and clarity. Beyond entertainment, this endeavor provides a practical application of programming methodologies, showcasing the structured approach required to design and implement interactive systems effectively.

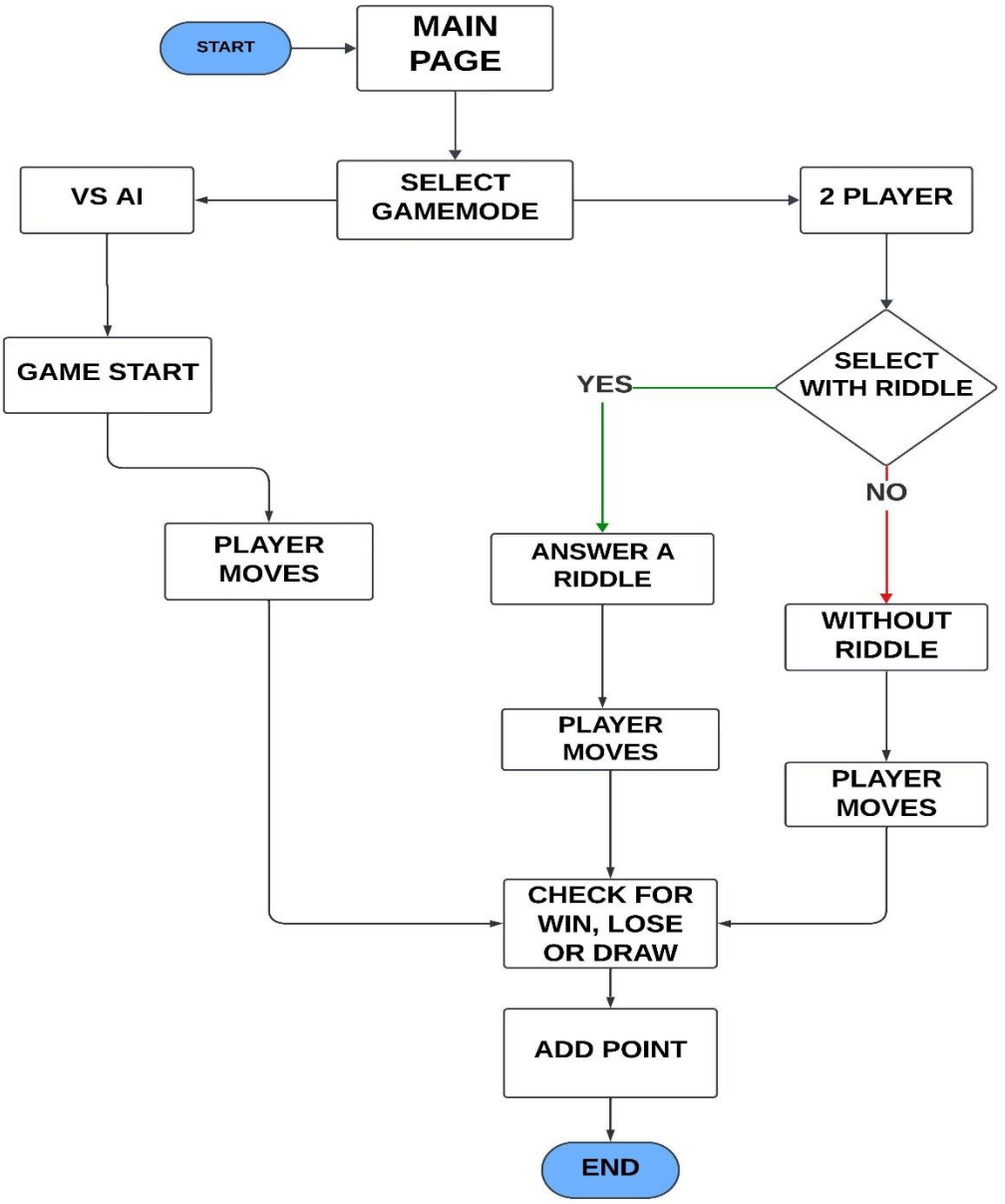
II. Objectives

- To design a user-friendly console-based Tic Tac Toe game.
- To implement OOP principles in designing the game's structure.
- To provide an interactive experience for players, with clear instructions and valid input checks.
- To demonstrate the use of algorithms for win detection and game flow management.

III. Scope and Delimitation of the Study

The project focuses on implementing a Tic Tac Toe game for two players. The game is designed for execution in a console environment. It supports basic functionalities, including board display, player input, and result evaluation (win, lose, or draw). The study does not extend to developing online multiplayer functionalities.

IV. Flowchart



The flowchart illustrates the logical flow of the Tic Tac Toe game:

i. The Game Start.

ii. Select in the Main Page if you want to play VS AI or 2 Player. In the Main Page the option button is included.

iii. If the Player select VS AI, the opponent is the computer.

iv. If the Player select 2 Player, he/she can choose either of the choices if its with riddle or without riddle.

v. After the player moves, check if its win, lose or draw.

vi. After checking the point will be added.

vii. End. The player can exit the game or continue to play.

V. Gantt Chart

TASK	October				November				December			
	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4	Week 1	Week 2	Week 3	Week 4
SYSTEM DESIGN												
GAME LOGIC												
UI DESIGN												
TESTING AND DEBUGGING												
MODE INTEGRATION												

October week 1-2: Creating the additional design.

Week 2-3: Adding the principle of the VS AI.

Week 3-4: Fixing the GUI of the board in the VS AI.

Week 4-November week 2: Fixing the game logic of the VS AI.

Week 2-4: Creating the Additional design of the game.

Week 4- December week 1: creating the Mode of the 2 player the Tic-Tac-Toe with riddle and without riddle.

Week 2: Fixing the bug of the Riddle.

VI. System Design

Fig 1.

The Figure 1 shows the menu page of the game Tic Tac Toe.

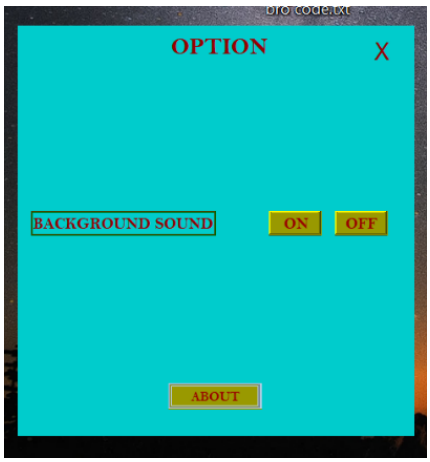


Fig 2.

This figure shows the option of the game Tic Tac Toe game. This include the button for the sound effects of the game and the “About” button for the instruction.

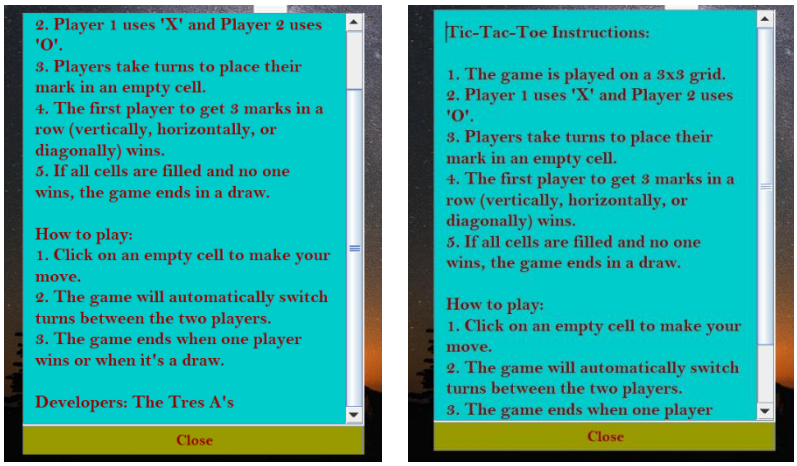


Fig 3.

This figure shows the instruction of the game for a guide for the new players.

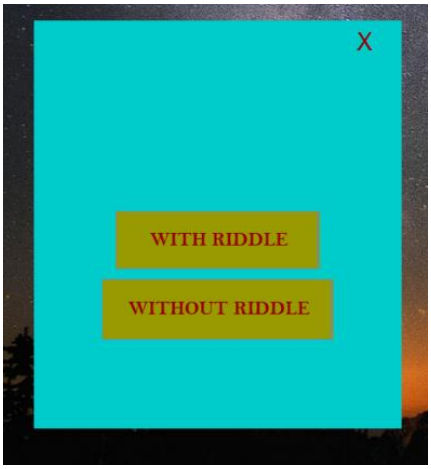


Fig 4.

It shows the mode of the 2 player if they want to play with the riddle or without the riddle.

Fig 5.

This is the Board of the 2 player with a riddle, it shows the input of the player for the riddle and the scoring of the players.

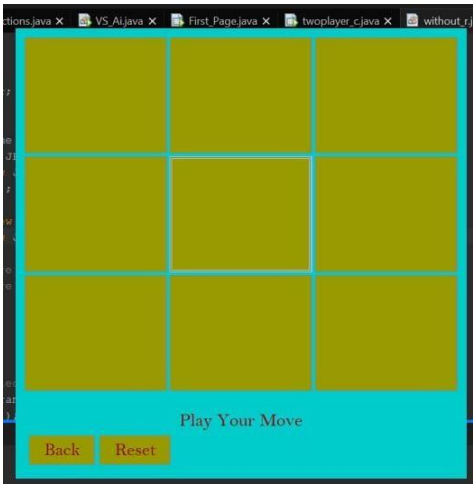


Fig 6.

This figure is the board of the VS AI and the 2 player without a riddle, they have the same design for the consistency of the system or the game Tic Tac Toe.

VII. Conclusion

The development of the Tic Tac Toe game demonstrates the practical application of object-oriented programming principles. By focusing on modular design, the project ensures code reusability, scalability, and clarity, providing a solid foundation for future enhancements.

VIII. Recommendation

For future developments, it is recommended to:

1. Implement a graphical user interface (GUI) for improved user interaction.
2. Introduce a match history to see the last move of the opponent.
3. Enable online multiplayer functionality using network programming.