Project Proposal:

Project Title:

- **Tic-Tac-Toe** Game Implementation.

Group Members:

- **Name**: Faique Mehmood
- **Student ID**: 24K-0736
- **Name**: Muhammad Umar Bin Zahid
- **Student Id**: 24K-0989
- **Name**: Ali Muhammad
- **Student Id**: 24K-0753

Submission Date:

[Date of Submission]

1. Executive Summary:

Overview:

This project aims to implement two versions of the classic Tic-Tac-Toe game in C++: one where two players compete against each other, and another where person competes with the computer. The project explores basic game development concepts, user input handling, graphical user interface and random number generation for the computer's moves. The goal was to create an engaging and interactive game that adheres to the traditional rules of Tic-Tac-Toe while providing a user-friendly experience.

Key Findings:

* Successfully implemented a functional Tic-Tac-Toe game for both single-player (against the Computer) and two player mode.
* Demonstrated the use of arrays, loops, and conditional statements in C++.
* Implemented basic game logic to check for wins, draws, and valid moves.

* To enhance the user experience, we plan to implement a graphical user interface using a library such as SFML. This will replace the current text-based interface with a more visually engaging and interactive design.

* Ensured the game handles user inputs correctly and provides clear feedback.

2. Introduction:

Background:

Tic-Tac-Toe, also known as Noughts and Crosses, is a timeless two-player game where players take turns marking spaces in a 3x3 grid. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game. This project was undertaken to gain hands-on experience in game development using C++, a powerful and versatile programming language. By implementing this game, we aimed to understand fundamental programming concepts such as user input handling, game logic, and random number generation.

Project Objectives:

* Develop a Tic-Tac-Toe game in C++ that can be played by one player against the computer.

* Develop a Tic-Tac-Toe game in C++ that can be played by two players against each other.

* Ensure the game handles user inputs correctly and checks for valid moves.

* Display the game board and results clearly to the users.

* Implement a simple AI for the computer player in the single-player mode.

3. Project Description:

Scope:

The project includes:

* A 3x3 game board displayed using a 5x5 character array for visual representation.

* Functions to handle user inputs, check for wins, and display the game board.

* Random move generation for the computer in the single-player mode.

* Graphical user interface (GUI) implementation.

Technical Overview:

> ****Programming Language**:** C++.

> ****Compiler**:** g++ (or any standard C++ compiler).

> ****Development Environment****: Any standard text editor or IDE (e.g., Visual Studio Code)

4. Methodology:

Approach

> ****Weekly Planning**:** Meetings to discuss progress and address any issues.

> **Division of Tasks:** Each group member was assigned specific tasks such as implementing game logic, handling user inputs, and displaying the game board.

Roles and Responsibilities:

- *Faique Mehmood* : Implemented the game logic, handled user inputs.
- *Muhammad Umar* : Implemented the Graphical User Interface(GUI), and wrote Project report and Project proposal.
- *Ali Muhammad* : Debugging code and fix the errors.

5. Expected Outcomes:

Deliverables:

- A fully functional Tic-Tac-Toe game with a user-friendly GUI.
- A project report detailing the development process and outcomes.
- User instructions for playing the game.

Relevance:

This project demonstrates the use of OOP concepts in game development and showcases the integration of a GUI to enhance user experience. It connects with ICT topics such as data organization, simple programming, and user interface design.

6. Resources Needed:

Software:

- C++ Compiler (g++)
- SFML Library
- Visual Studio Code or any standard IDE

Other Resources:

- Online tutorials for SFML.
- Support from the instructor.