* **Introduction:**

In the below stated project I had to make a tic tac toe game using java programming language.Tic Tac Toe is a most common and typical two-player game which includes filling a 3x3 grid with X’s and O’s in different turns getting three in a row. The aim of this project is to achieve a digital version of the game using Java programming language and tools.The purpose of this project are to build a functioning Tic Tac Toe game which provides two players to play oppose to one another. The goals are to develop a game with even rules and winning circumstances, implementing a user interface that is spontaneous and uncomplicated to work and gives an enjoyable experience for players. In this game we have four classes they are Game, GameBoard, Player,TicTacToeGUI respectively.

In the Game class it contains X and O record button.it keeps the records the button of the x and o .it has the winning conditions.

In GameBoard class it has the record of how many buttons are used.and all of the buttons are used in array form.

Player class defines the turn of the player also has an instance variable.

The TicTacToeGUI class represents a Tic Tac Toe game with a graphical user interface. It has nine JButton objects and instance variables for GameBoard, Game, and Player obj package com.mycompany.tictacgui;

* **Implemation:**

The programming language used to develop the game is Java, and the game is implemented using the Swing framework. Swing is a GUI toolkit for Java, and it provides the necessary tools for creating user interfaces, handling user input, and displaying graphics. The game is implemented using a series of classes that handle user input, game logic, and the visual display of the game board. Loops are also used to iterate through the game board and check for win conditions.

* **Source code:**

import java.awt.Dimension;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JOptionPane;

public class TicTacToeGUI extends JFrame implements ActionListener {

Player player;

GameBoard gameboard;

public Game game;

JButton b0, b1, b2, b3, b4, b5, b6, b7, b8;

public TicTacToeGUI(Game game, GameBoard gameboard, Player player) {

this.game = game;

this.gameboard = gameboard;

this.player = player;

b0 = new JButton();

b1 = new JButton();

b2 = new JButton();

b3 = new JButton();

b4 = new JButton();

b5 = new JButton();

b6 = new JButton();

b7 = new JButton();

b8 = new JButton();

setTitle("Tic Tac Toe");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(300, 300);

setLayout(new GridLayout(3, 3));

b0.setPreferredSize(new Dimension(100, 100));

b0.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[0] == 0) {

if (player.nowPlayer == 'X') {

b0.setText("X");

gameboard.moves++;

game.pXset[0] = 1;

gameboard.btnUsed[0] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b0.setText("O");

gameboard.moves++;

game.pOset[0] = 1;

gameboard.btnUsed[0] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b0, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b0);

b1.setPreferredSize(new Dimension(100, 100));

b1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[1] == 0) {

if (player.nowPlayer == 'X') {

b1.setText("X");

gameboard.moves++;

game.pXset[1] = 1;

gameboard.btnUsed[1] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b1.setText("O");

gameboard.moves++;

game.pOset[1] = 1;

gameboard.btnUsed[1] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b1, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b1);

b2.setPreferredSize(new Dimension(100, 100));

b2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[2] == 0) {

if (player.nowPlayer == 'X') {

b2.setText("X");

gameboard.moves++;

game.pXset[2] = 1;

gameboard.btnUsed[2] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b2.setText("O");

gameboard.moves++;

game.pOset[2] = 1;

gameboard.btnUsed[2] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b2, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b2);

b3.setPreferredSize(new Dimension(100, 100));

b3.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[3] == 0) {

if (player.nowPlayer == 'X') {

b3.setText("X");

gameboard.moves++;

game.pXset[3] = 1;

gameboard.btnUsed[3] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b3.setText("O");

gameboard.moves++;

game.pOset[3] = 1;

gameboard.btnUsed[3] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b3, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b3);

b4.setPreferredSize(new Dimension(100, 100));

b4.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[4] == 0) {

if (player.nowPlayer == 'X') {

b4.setText("X");

gameboard.moves++;

game.pXset[4] = 1;

gameboard.btnUsed[4] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b4.setText("O");

gameboard.moves++;

game.pOset[4] = 1;

gameboard.btnUsed[4] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b4, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b4);

b5.setPreferredSize(new Dimension(100, 100));

b5.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[5] == 0) {

if (player.nowPlayer == 'X') {

b5.setText("X");

gameboard.moves++;

game.pXset[5] = 1;

gameboard.btnUsed[5] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b5.setText("O");

gameboard.moves++;

game.pOset[5] = 1;

gameboard.btnUsed[5] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b5, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b5);

b6.setPreferredSize(new Dimension(100, 100));

b6.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[6] == 0) {

if (player.nowPlayer == 'X') {

b6.setText("X");

gameboard.moves++;

game.pXset[6] = 1;

gameboard.btnUsed[6] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b6.setText("O");

gameboard.moves++;

game.pOset[6] = 1;

gameboard.btnUsed[6] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b6, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b6);

b7.setPreferredSize(new Dimension(100, 100));

b7.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[7] == 0) {

if (player.nowPlayer == 'X') {

b7.setText("X");

gameboard.moves++;

game.pXset[7] = 1;

gameboard.btnUsed[7] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b7.setText("O");

gameboard.moves++;

game.pOset[7] = 1;

gameboard.btnUsed[7] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b7, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b7);

b8.setPreferredSize(new Dimension(100, 100));

b8.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (gameboard.btnUsed[8] == 0) {

if (player.nowPlayer == 'X') {

b8.setText("X");

gameboard.moves++;

game.pXset[8] = 1;

gameboard.btnUsed[8] = 1;

if (gameboard.moves >= 5 && game.pXWin()) {

playAgainConfirm('X');

}

player.nowPlayer = 'O';

} else {

b8.setText("O");

gameboard.moves++;

game.pOset[8] = 1;

gameboard.btnUsed[8] = 1;

if (gameboard.moves >= 5 && game.pOWin()) {

playAgainConfirm('O');

}

player.nowPlayer = 'X';

}

} else {

JOptionPane.showMessageDialog(b8, "Button is already used");

}

if (gameboard.moves == 9) {

playAgainConfirm('D');

}

}

});

add(b8);

setVisible(true);

}

public void playAgainConfirm(char x) {

if (x == 'X') {

String winMessage = "Player X wins the game!\n";

int dialogResult = JOptionPane.showOptionDialog(null, winMessage+"Do you want to Play Again?", "Confirmation", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE, null, null, null);

if (dialogResult == JOptionPane.YES\_OPTION) {

resetGame();

} else {

System.exit(0);

}

} else if (x == 'O') {

String winMessage = "Player O wins the game!\n";

int dialogResult = JOptionPane.showOptionDialog(null, winMessage+"Do you want to Play Again?", "Confirmation", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE, null, null, null);

if (dialogResult == JOptionPane.YES\_OPTION) {

resetGame();

} else {

System.exit(0);

}

} else if (x == 'D') {

String winMessage = "It's a DRAW !!\n";

int dialogResult = JOptionPane.showOptionDialog(null, winMessage+"Do you want to Play Again?", "Confirmation", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE, null, null, null);

if (dialogResult == JOptionPane.YES\_OPTION) {

resetGame();

} else {

System.exit(0);

}

}

}

public void resetGame() {

gameboard.resetGameBoard();

game.resetGame();

player.resetPlayer();

blankBtns();

}

public void blankBtns() {

b0.setText("");

b1.setText("");

b2.setText("");

b3.setText("");

b4.setText("");

b5.setText("");

b6.setText("");

b7.setText("");

b8.setText("");

}

@Override

public void actionPerformed(ActionEvent e) {

throw new UnsupportedOperationException("Not supported yet.");

}

public static void main(String[] args) {

Game game = new Game();

GameBoard gameboard = new GameBoard();

Player player = new Player();

new TicTacToeGUI(game, gameboard, player);

}

}ects.It also has the main class in it.

* **Challenges and Solutions:**

While working with the Tic-Tac-Toe game project I have faced some problems and errors. One of the most potential obstacles of them was the idea building for the game as per instructions. The game consists of 9 buttons. So there were possibilities that repeating codes for each buttons could have missing statements as well. The idea building for the game logic and implementing them into the project was also a challenging part.the main obstacle that I faced was when first I call main class then the file did not show up any result but later when I created object in next three classes then the result showed up.so from this I learned we should be more careful and the code should be debugging before we use it.

* **Discussion :**

Completing this assignment, I have understood and got a clear visions of the java programming language. Even though it was a classic game project but I made it a bit informative because any user can easily get the idea of how to the game would be implemented with the help of java language and GUI. The assignment was a great opportunity to practice Java programming and user interface design I faced several challenges during development, but through practice.However,I was able to overcome them and create a functional and enjoyable Tic Tac Toe game.