

Carlos Eduardo Cordon Hernandez  
201504427  
Practica 1

# Manual Técnico

---

# Introducción

---

La práctica consiste en el desarrollo de un juego de Totito como un programa de consola con ciertas modificaciones para agregar complejidad al juego. Se podrá definir un tamaño variable para el tablero de  $n \times n$  ( $3 \leq n \leq 10$ ) y asociado al tamaño del tablero también variará el número de jugadores.

El tablero de juego será manejado estrictamente haciendo uso de memoria estática, declarando una matriz de dos dimensiones de tamaño  $10 \times 10$ . Para cada juego se utilizará únicamente una sección de dicha matriz, dependiendo del tamaño de tablero que haya seleccionado el usuario.

# Requerimientos Del Sistema

---

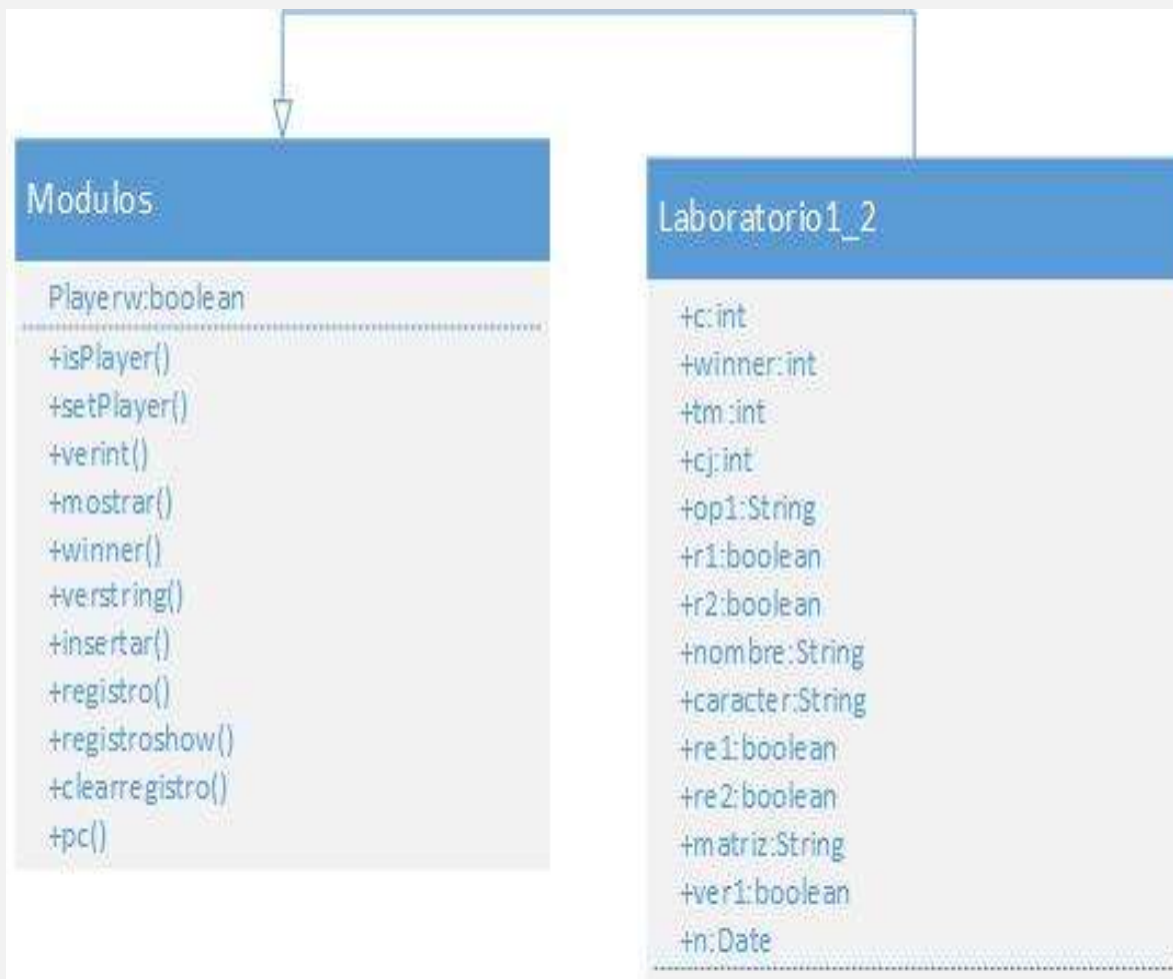
- El sistema actual tiene la capacidad de jugar un totito de 3x3 hasta un totito de 10x10 únicamente.
- El tamaño del totito restringe la capacidad o pone una limitante a la cantidad de jugadores ejemplo (totito 3x3 únicamente para 2 jugadores)
- El esquema del juego está diseñado para ganar conectando un línea tanto en diagonal, horizontal y/o vertical.
- Las opciones para agregar campos están definidas en posición de coordenada siempre y cuando respete el tamaño de la matriz y que sus condiciones iniciales que la primera fila como tanto la primera columna todo esto inicia en 0.
- El sistema permite almacenar registros de los ganadores y sus rivales definiendo rival, carácter ganador y hora en que se dio fin a la partida.
- El usuario no tiene permitido sobrescribir en la matriz tablero sobre alguna posición rival o propia.

# Requerimientos De Usuario

---

- Definiendo la cantidad de jugadores el sistema le permite al usuario visualizar la opción que quiere de juego mediante siempre estén disponibles
- En base a la cantidad de jugadores el sistema le permite al usuario definir un nombre personalizado al campo de jugadores y con su propio carácter
- Cada turno el jugador visualiza su posición y a la vez la del rival aun así el juego haya terminado como empate o tomando en cuenta un ganador en cada instante se ve el aspecto del tablero.
- Una vez se defina un ganador registra al ganador y muestra un mensaje con sus datos
- En dado caso quede en empate el juego se da por terminado
- Todo juego se puede jugar una vez pero siempre esta latente de volver a jugar otro juego con las mismas o distintas característica

# Diagrama De Clases



# Diccionario De Clases

---

## Clase: Modulos

Es la clase "cerebro" donde todos los método inteligentes están almacenadas en ella para interactuar con los atributos de otras clases.

### Atributos

Plawerw	Almacena el estado del juego referente a si hay un ganador en el juego
---------	--

### Métodos

isPlayerw	Obtiene el valor de variable Playerw
setPlayerw	Agrega valor a la variable Playerw
Verint	Verifica que lo que inserte sea un numero entero positivo
Mostrar	Muestra el tablero
Winner	Verifica si el jugador en turno gana
Verstring	Verifica que el usuario meta la sintaxis de la forma correcta
Insertar	Inserta los campos validos en la matriz
Registro	Registra al ganador y sus datos en un documento de extensión .txt
Registroshow	Muestra los campos o valores de mi registro
Clearregistro	Limpia mi registro
pc	Metodo para jugar contra la pc

## Clase: Laboratorio1\_2

Es la clase que permite el juego y toda la interacción con el usuario

### Atributos

C	Variable contador de tipo entero
Winner	Variable booleana que verifica estado actual de la partida
Tm	Tamaño de la matriz
Cj	Cantidad de jugadores
Op1	Variable de tipo string para opciones de teclado
R1	Variable booleana para ciclos
R2	Variable booleana para ciclos
Nombre	Arreglos que almacena nombres
Caracteres	Arreglo que almacena caracteres
Re1	Variable booleana para ciclos
Re1	Variable booleana para ciclos
Matriz	Matriz que funciona como tablero
Ver1	Variable booleana para ciclos
N	Variable de tipo Date para datos de sistema horario.