

Laboratorio 2 Arreglos y Excepciones

Marco Soloj 25701

1.1 Requisitos funcionales del sistema

- El user debe poder ingresar a 2 jugadores para el juego de memoria
- El tablero debe ser del tamaño que el jugador diga (solo pares debido a que si es un número impar no hay parejas suficientes, no cuadran los números).
- El jugador debe poder seleccionar 2 cartas para intentar adivinar si son pares indicando su fila y columna
- Se deben revelar las cartas seleccionadas y comprobar si son una pareja, si son pareja dar un punto al jugador que haya hecho el match
- Se debe poder jugar múltiples veces reiniciando el tablero, de distintos tamaños y si no desean jugar más mostrar quien es el ganador.

1.2 Clases necesarias y su propósito

Clase	Propósito
MainLab2	Clase principal que ejecuta el programa y llama al método Game() de VistaLab2.
VistaLab2	Gestiona toda la interacción con el usuario: pide datos de entrada, muestra el tablero, controla los turnos y maneja el flujo del juego.
ControladorLab2	Administra la lógica del juego: validación de jugadas, control de puntos, reinicio de partida y determinación del ganador.
CartaLab	Representa una carta del tablero con su símbolo, estado (revelada u oculta) y si ya fue emparejada.
TableroLab	Contiene todas las cartas del juego. Se encarga de generarlas en pares, mezclarlas, y administrar su estado.
JugadorLab	Representa a cada jugador con su nombre, puntaje actual y la cantidad de partidas ganadas.

1.3 Atributos de cada clase

Clase: CartaLab

Atributo	Tipo de dato	Visibilidad	Propósito
simbolo	String	- (privado)	Almacena el símbolo único que identifica la carta.
Revelada	boolean	- (privado)	Indica si la carta está descubierta temporalmente.

Matched	boolean	- (privado)	Indica si la carta ya ha sido emparejada correctamente.
---------	---------	-------------	---

Clase: JugadorLab

Atributo	Tipo de dato	Visibilidad	Propósito
nombre	String	- (privado)	Nombre del jugador.
puntos	int	- (privado)	Puntos obtenidos en la partida actual.
gamesGanados	int	- (privado)	Número de partidas ganadas a lo largo del juego.

Clase: TableroLab

Atributo	Tipo de dato	Visibilidad	Propósito
cartas	ArrayList<CartaLab>	- (privado)	Lista que contiene todas las cartas del tablero.
tamaño	int	- (privado)	Número total de cartas en el tablero.
n	int	- (privado)	Tamaño del tablero (n x n).

Clase: ControladorLab2

Atributo	Tipo de dato	Visibilidad	Propósito
tablero	TableroLab	- (privado)	Representa el tablero del juego.
jugador1	JugadorLab	- (privado)	Primer jugador del juego.
jugador2	JugadorLab	- (privado)	Segundo jugador del juego.

1.4 Métodos de cada clase

Clase: VistaLab2

Método	Parámetros	Visibilidad	Propósito
Game()	ninguno	+ (público)	Controla todo el flujo del juego, interacción con los jugadores y turnos.
displayTablero(tablero: TableroLab)	TableroLab	- (privado)	Recorre el tablero y muestra las cartas

			dependiendo de su estado (oculta, revelada o emparejada).
--	--	--	---

Clase: ControladorLab2

Método	Parámetros	Visibilidad	Propósito
jugar (jugador: JugadorLab, carta1: CartaLab, carta2: CartaLab) : String	JugadorLab, CartaLab, CartaLab	+ (público)	Valida una jugada, revela cartas, asigna puntos y gestiona coincidencias.
nuevoJuego(n:int)	int	+ (público)	Reinicia el tablero y restablece los puntos de los jugadores.
Victoria() : String	ninguno	+ (público)	Determina el ganador comparando los puntos de ambos jugadores.

Clase: TableroLab

Método	Parámetros	Visibilidad	Propósito
getCarta(fila:int, columna:int) : CartaLab	int, int	+ (público)	Devuelve la carta en la posición indicada.
FinJuego() : boolean	ninguno	+ (público)	Verifica si todas las cartas han sido emparejadas y el juego terminó.

Clase: CartaLab

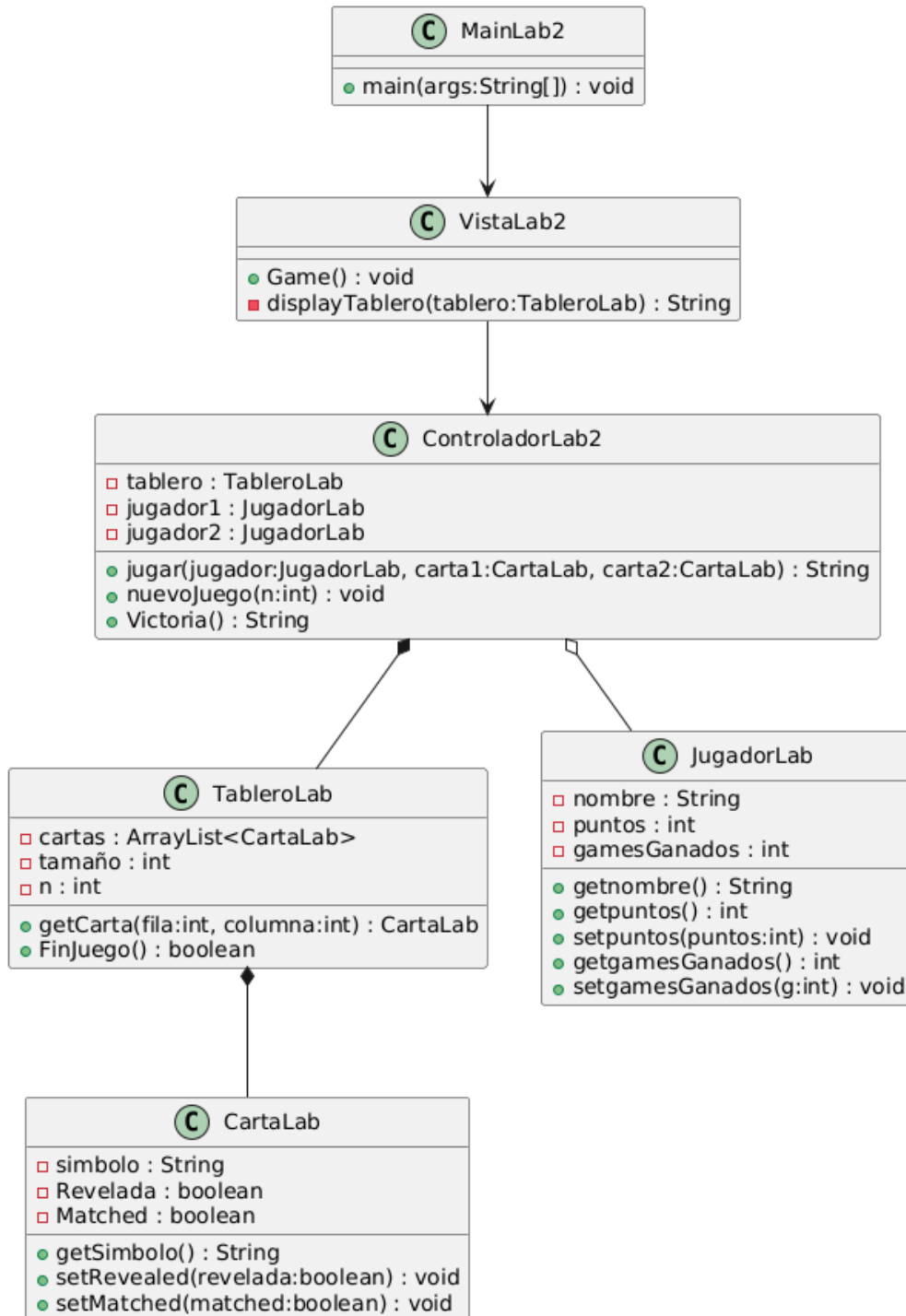
Método	Parámetros	Visibilidad	Propósito
getSimbolo() : String	ninguno	+ (público)	Devuelve el símbolo de la carta.
setRevealed(revelada:boolean) : void	boolean	+ (público)	Cambia el estado de revelada de la carta.
setMatched(matched:boolean) : void	boolean	+ (público)	Marca la carta como emparejada.

Clase: JugadorLab

Método	Parámetros	Visibilidad	Propósito
getnombre() : String	ninguno	+ (público)	Devuelve el nombre del jugador.
getpuntos() : int	ninguno	+ (público)	Obtiene los puntos del jugador.
setpuntos(puntos:int) : void	int	+ (público)	Asigna nuevos puntos al jugador.
getgamesGanados() : int	ninguno	+ (público)	Obtiene la cantidad de juegos ganados.
setgamesGanados(g:int) : void	int	+ (público)	Actualiza la cantidad de juegos ganados.

1.5 Diagrama

Juego de Memoria - Diagrama de Clases



1.6 Enlace Github

Enlace: <https://github.com/Maquito365/Lab2-Marco-Soloj>

<https://github.com/Maquito365/Lab2-Marco-Soloj>