

Kevin Jordan Alzate – 2228507
<kevin.jordan@correounivalle.edu.co>
Junior Cantor Arévalo – 2224949
<junior.cantor@correounivalle.edu.co>
Edier Johan Castro Vargas – 2224517
<edier.castro@correounivalle.edu.co>

Diseño de clases

Batalla: Proporciona la funcionalidad necesaria para gestionar las batallas en el juego, actualizando los tableros de los jugadores de acuerdo con las casillas ocupadas por las naves del oponente.

GUIBatallaNaval: Se define GUI base para agregar los botones y paneles, también se establecen todos los escuchas

GUIPosicionesCPU: Gestiona la visualización de las posiciones del oponente en la GUI del juego de Batalla Naval, realizar ataques del oponente y distribuir aleatoriamente la flota del oponente en su tablero.

Header: Esta clase establece la etiqueta del título de nuestro proyecto

MatricesTableros: Almacena la información sobre las casillas ocupadas por barcos y permitiendo la manipulación de estos datos a través de métodos específicos.

MostrarBarcos: Realiza la lógica y las verificaciones adecuadas para esperar que en el tablero del usuario se organicen los barcos de forma correcta para pintarlos y que se puedan visualizar.

MostrarBarcosCPU: Realiza la lógica y las verificaciones adecuadas para esperar que en el tablero de la cpu se organicen los barcos de forma correcta para pintarlos y que se puedan visualizar.

PanelAsignaciones: Se establece los métodos que permiten organizar nuestro panel con los respectivos barcos, se modifica posición y orientación. También recoge información del juego, da la señal de asignar el turno y permite obtener un botón de explicación.

PanelCPUTablero: Crea un panel donde se agregarán las matrices establecidas para los tableros posición, principal del oponente y los retorna.

PanelMatrices: Crea un panel donde se agregarán las matrices establecidas para los tableros posición, principal del usuario y los retorna.