

750014C FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A EVENTOS - 2024-2

ENUNCIADO MINI PROYECTO #3

Descripción: En este mini proyecto se desarrollará el juego llamado "Batalla Naval", el cual es un juego de estrategia donde dos jugadores (humano y máquina) compiten por hundir la flota del oponente. Cada jugador coloca sus barcos en un tablero de coordenadas y luego intenta adivinar la ubicación de los barcos del oponente para hundirlos. Teniendo en cuenta que se jugará contra la máquina el juego tendrá 2 tableros de 10x10:

- **Tablero de posición:** Representa el territorio del jugador humano, en él se distribuye su flota antes de comenzar la partida y sólo será de observación. Verá la posición de sus barcos y los disparos de su oponente en su territorio, pero no podrá realizar ningún cambio ni disparo en él.
- **Tablero principal:** Representa el territorio del jugador máquina, donde tiene desplegada su flota. Será aquí donde se desarrollen los movimientos (disparos) del jugador humano tratando de hundir los barcos enemigos. Este tablero aparecerá en la pantalla del jugador una vez comience la partida y en él quedarán registrados todos sus movimientos, reflejando tanto los disparos al agua como los barcos tocados y hundidos en tiempo real.

Cada jugador tiene una flota de 10 barcos de diferente tamaño, ubicados de manera horizontal o vertical, por lo que cada uno ocupará un número determinado de casillas en el tablero distribuidos de la siguiente manera:

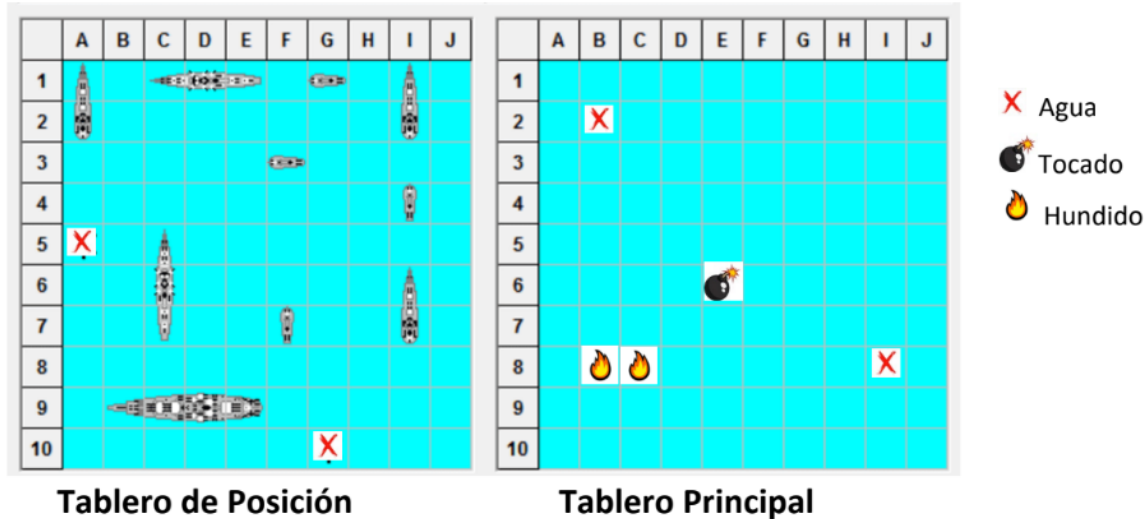
- **1 portaaviones:** ocupa 4 casillas.
- **2 submarinos:** ocupan 3 casillas cada uno.
- **3 destructores:** ocupan 2 casillas cada uno.
- **4 fragatas:** ocupan 1 casilla cada uno.

En este juego se tiene las siguientes terminologías y movimientos:

- **Agua:** cuando se dispara sobre una casilla donde no está colocado ningún barco enemigo. En el tablero principal del jugador aparecerá una X. Pasa el turno a su oponente.
- **Tocado:** cuando se dispara en una casilla en la que está ubicado un barco enemigo que ocupa 2 o más casillas, se destruirá sólo una parte del barco. En el tablero del jugador aparecerá esa parte del barco con una marca indicativa de que ha sido tocado. El jugador vuelve a disparar.
- **Hundido:** si se dispara en una casilla en la que está ubicado una fragata (1 casilla) u otro barco con el resto de las casillas tocadas, el barco se hundirá, es decir, se ha eliminado ese barco del juego. Aparecerá en el tablero principal del jugador, el barco completo con la marca indicativa de que ha sido hundido. El jugador puede volver a disparar, siempre y cuando no haya hundido toda la flota de su enemigo, en cuyo caso habrá ganado.

750014C FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A EVENTOS - 2024-2

Para este juego, se debe emular el juego del computador (quien será el oponente). Para ello, se podrán aleatoriamente los barcos en el tablero del territorio, siguiendo las reglas y se selecciona al azar cada casilla de tiro. Aquí puedes ver un ejemplo de un escenario de juego (Vista del jugador humano).



Para efectos de verificar que la jugabilidad esté correcta se debe proveer una opción que permita visualizar el tablero de posición del oponente (computador).

Objetivo general: Desarrollar el juego "Batalla Naval," implementando una versión digital donde un jugador humano compite contra una máquina. El proyecto debe incluir una interfaz gráfica intuitiva y atractiva, con dos tableros de juego: uno para la posición de los barcos del jugador humano y otro para los ataques al territorio del jugador máquina.

Objetivos específicos:

- Crear una interfaz gráfica de usuario intuitiva con aspectos básicos de UX y UI.
- Emplear interfaces, clases adaptadores e internas.
- Implementar eventos de mouse.
- Emplear la arquitectura Modelo-Vista-Controlador en el código fuente.
- Documentar el código fuente usando Javadoc.
- Emplear layouts en la interfaz gráfica de usuario.
- Implementar una estructura de datos.
- Gestionar el proyecto por medio de git y alojarlo en GitHub.
- Implementar archivos planos (nickname y cantidad de barcos hundidos) y serializables (tableros).
- Implementar figuras 2D para los barcos y elementos del juego (agua, tocado y hundido).
- Implementar excepciones de Java marcadas, no marcadas y propias.

750014C FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A EVENTOS - 2024-2

Tecnologías y herramientas para desarrollar el sol eclipsado: Java, JavaFX, Scene Builder, IntelliJ IDEA, Git y GitHub.

Cantidad de integrantes por grupo: Tres estudiantes.

Historias de usuario: A continuación, se presentan las funcionalidades del mini proyecto.

HU-1 Colocación de barcos
Como jugador humano, quiero poder colocar mis barcos en el tablero de posición para establecer mi flota antes de comenzar la partida.
Criterios de aceptación: <ul style="list-style-type: none">El jugador puede seleccionar y arrastrar barcos en el tablero de posición.Los barcos pueden ser colocados horizontal o verticalmente.El sistema debe validar que los barcos no se superpongan ni salgan del tablero.Una vez colocados, los barcos no pueden ser movidos ni modificados.
Definición de hecho: <ul style="list-style-type: none">Todos los barcos están colocados correctamente en el tablero de posición sin superposiciones ni errores de ubicación.Los barcos son visibles en la interfaz, y el tablero refleja con precisión la disposición de la flota del jugador.

HU-2 Realización de disparos
Como jugador humano, quiero poder disparar en el tablero principal del oponente (máquina) para intentar hundir los barcos enemigos.
Criterios de aceptación: <ul style="list-style-type: none">El jugador puede seleccionar una celda en el tablero principal para disparar.El sistema indica si el disparo es agua, tocado o hundido.Los resultados de los disparos se muestran en el tablero en tiempo real.El turno del jugador termina si el disparo es agua; si es tocado o hundido, puede seguir disparando.
Definición de hecho: <ul style="list-style-type: none">Los disparos se registran correctamente en el tablero principal con las marcas adecuadas (X para agua, indicador de tocado, y símbolo de hundido).El sistema maneja correctamente el cambio de turnos según los resultados de los disparos.

HU-3 Visualización del tablero de posición del oponente
Como profesor, quiero poder visualizar el tablero de posición del oponente (máquina) para verificar que la colocación de los barcos y la jugabilidad sean correctas.
Criterios de aceptación: <ul style="list-style-type: none">Existe una opción o función que permita mostrar el tablero de posición del oponente.La visualización debe ser clara y mostrar la disposición de la flota del oponente.Esta opción debe estar disponible únicamente para fines de verificación y no durante el juego normal.

750014C FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A EVENTOS - 2024-2

Definición de hecho:

- El tablero de posición del oponente se muestra correctamente con todos los barcos ubicados.
- La interfaz refleja con precisión la disposición de la flota del oponente.

HU-4 Implementación de la inteligencia artificial de la máquina

Como jugador humano, **quiero** enfrentarme a una máquina que juegue de manera autónoma **para** poder disfrutar de una partida de Batalla Naval en solitario.

Criterios de aceptación:

- La máquina coloca sus barcos aleatoriamente siguiendo las reglas del juego.
- La máquina selecciona casillas de disparo de manera aleatoria durante su turno.
- La máquina responde adecuadamente a los disparos del jugador humano, indicando agua, tocado o hundido.

Definición de hecho:

- La máquina coloca correctamente sus barcos sin superposiciones y dentro del tablero.
- Los disparos de la máquina se realizan y se registran de manera coherente con las reglas del juego.

HU-5 Guardado automático del juego

Como jugador, **quiero** que cada vez que se realice una jugada (por parte del jugador o de la máquina) el estado del juego se guarde automáticamente, **para** que al volver a ingresar pueda continuar exactamente desde donde me encontraba, sin perder progreso.

Criterios de aceptación:

- El sistema guarda automáticamente el estado del juego tras cada jugada, ya sea realizada por el jugador o por la máquina.
- El estado guardado incluye:
 - **Tablero:** El estado completo del tablero actual, serializado para facilitar su recuperación.
 - **Jugador:** Nickname del jugador y cantidad de barcos hundidos.
 - **Máquina:** Estado del tablero de la máquina, si aplica.
- Los archivos de guardado deben cumplir con los siguientes requisitos:
 - **Archivos serializables:** para el estado del tablero, de modo que pueda recuperarse fácilmente al reanudar el juego.
 - **Archivos planos** para almacenar información como el nickname del jugador y el estado de los barcos hundidos.
- Al volver a iniciar sesión, el sistema carga automáticamente el archivo de guardado más reciente y restaura el estado exacto en el que se encontraba el jugador.

Definición de hecho:

- Se valida que, al cerrar y reabrir el juego, este continúa desde el estado exacto en que se encontraba, sin pérdida de datos.
- Se verifica que los archivos generados contienen la información mencionada (tablero serializado, nickname y barcos hundidos).