

Proyecto Final Programación 2

Grupo 7

Facultad de Ingeniería y Tecnologías

Integrantes:

José Varela

Nicolás Lorenzo

Manuel Baldomir

Juan Campos

Docentes:

Matias Olave

María Parapar

Felipe López

ÍNDICE

Funcionalidades añadidas.....	3
Funcionamiento	5
Roles	8

Funcionalidades añadidas

Se nos pide añadir nuevas funcionalidades al proyecto, por lo tanto se optó por agregar algunos modos de juegos diferentes al juego clásico y además agregarle algunas funcionalidades extras que se muestran a continuación.

- Bomb
- TimeTrial
- Challenge
- Quickchat
- Ranking

Bomb: El objetivo de esta clase es afectar directamente la forma básica de ataque que tienen los usuarios mientras juegan a la guerra naval. Al aplicar este modo, luego de que alguno de los jugadores erre 3 tiros seguidos en sus turnos correspondientes, su próximo disparo no será de solo 1 slot sino que será un cuadrado de dimensión 3x3.

TimeTrial: Esta clase afecta directamente en la modalidad de juego y en quien gana la partida. No solo gana quien hunde todos los barcos enemigos, sino también el que haya golpeado a más barcos enemigos en el tiempo límite que tendrá la partida contra reloj. Al finalizar el tiempo se verificará quien derribó y golpeó a más barcos enemigos y así se determinará quién ganó la partida si no fueron derribados todos los barcos. También afectará el tiempo que tiene cada jugador en realizar su ataque

.

Challenge: Tiene como objetivo realizar un reto con otro jugador y no solo jugar 1 partida entre sí, sino un mejor de 3.

QuickChat: Esta clase brinda la funcionalidad de poder intercambiar líneas rápidas de chat. Tendrá un pequeño repertorio de textos predefinidos que podrán ser utilizados por el jugador durante la partida para molestar al contrincante o simplemente ser amistoso.

Ranking: Esta funcionalidad permite a los usuarios visualizar un ranking global de quienes juegan a la guerra naval mediante nuestra aplicación. La información proporcionada será de juegos jugados y ganados distinguiendo cada modo de juego.

Funcionamiento

Para abordar este proyecto en el cual se debería crear o implementar un juego clásico llamado “Batalla Naval” se decidió realizar un diagrama UML para así tener un primer acercamiento a lo que sería la solución de este.

La solución debería tener varias clases para facilitar su funcionamiento desde el momento en que ingresa un usuario hasta que finaliza el juego.

Al comenzar, se debe mostrar un menú, en el cual el usuario pueda seleccionar alguna de las opciones disponibles, entre estas están las opciones de registrarse, jugar, salir, etc. En caso que el usuario ya esté registrado, le permite iniciar sesión ingresando el ID que se generó cuando creó el usuario y en caso de que no lo esté, se le pide que lo haga.

Una vez dentro, tendrá tres opciones, jugar, estadísticas y salir. La opción estadísticas le permite al jugador visualizar las estadísticas generales de los usuarios que hayan jugado y los juegos ganados por ellos. En cambio, si selecciona jugar se le solicita que seleccione el modo de juego que se quiera jugar y el usuario es asignado en una lista de espera para el modo de juego seleccionado. Cuando dos o más usuarios estén en la misma lista, es decir que estén esperando para ingresar al mismo modo de juego, se selecciona a 2 de ellos, el administrador genera una nueva instancia de juego, y comienzan a jugar en dicha sesión. A su vez se los elimina de la lista de espera en la que estaban y el administrador monitorea todas las partidas que están en juego sin importar el modo.

Una vez iniciada la sesión de juego, se le asigna a cada usuario 4 naves por defecto (Lancha, Submarino, Acorazado, Portaaviones), las cuales deberá posicionar en su tablero de dimensión 10x10. Habrán chequeos internos verificando si la posición seleccionada por el jugador es una posición válida para la nave seleccionada.

Luego de posicionar los barcos comienza la batalla naval. Al jugar existen 2 acciones básicas, una de ellas es atacar y la otra es mostrar los tableros. Al mostrar los tableros se verá el tablero propio con los ataques del oponente y los barcos del usuario, y el tablero del enemigo, en el cual solo serán visibles los ataques que el jugador haya realizado con distintivos de si acertó o erró a una nave enemiga.

Mientras estén jugando atacarán por turnos y también existe un tiempo límite para cada turno, es decir, un jugador no podrá demorar más de un tiempo prefijado ya que automáticamente pierde la instancia de ataque y se le da el turno a su oponente. Además tendrán la posibilidad de enviarse mensajes a través de la nueva función quick chat o solicitar una lista con todos los comandos disponibles para el juego.

Todas las interacciones entre los jugadores y el juego se darán a través del chat bot quien ejecutará la acción solicitada por el jugador.

En el momento que se hayan derribado todas las naves de alguno de los jugadores, la partida finalizará y mostrará por pantalla quién fue el ganador, también se le sumará una victoria al jugador correspondiente y se guardarán sus estadísticas. Luego se les consultará a los jugadores si quieren jugar otra partida entre ellos o no. Para que eso suceda ambos jugadores deben seleccionar que sí, caso contrario se los enviará al menú principal donde podrán elegir para jugar una nueva partida y buscar oponente o salir del juego. Al momento de retirar a ambos jugadores al menú principal se elimina la instancia de juego generada.

Ya explicado el juego “clásico”, existen algunas modificaciones o nuevos modos de juego añadidos a este para que el usuario quien lo va a jugar pueda seleccionar entre ellas cual prefiera.

Entre los nuevos modos de juego encontramos el modo “bomba” el cual es bastante particular porque como lo dice su nombre, afecta en el ataque del usuario, es decir, el modo de juego es totalmente igual al “clásico” pero al momento de atacar, en caso que el usuario no haya acertado determinada cantidad de disparos, se le asigna un disparo “bomba” el cual afecta más cantidad de posiciones brindando de esta forma una posibilidad mucho mayor de acertar en el disparo.

Por otro lado también nos encontramos con otro modo de juego diferente, este se denomina “TimeTrial”, el cual también se basa en el modo de juego “clásico” pero en esta modalidad el tiempo es un factor muy importante debido a que se establece una partida con un tiempo que ya esté predefinido y será limitado, al momento que comienza la partida el tiempo comienza a correr y si al finalizar el tiempo ninguno de los dos jugadores ganó, se establece como ganador el jugador que haya acertado más disparos en las naves del oponente. Esta es una modalidad de juego bastante interesante porque el juego se puede tornar un poco largo o extenso en caso de que los jugadores no acierten los disparos en las naves de los oponentes, en cambio con esta nueva funcionalidad se pueden hacer partidas más cortas y rápidas de jugar.

Finalmente se optó por agregar otra nueva funcionalidad a este juego la cual es de “challenge”, esta es bastante interesante o entretenida para el usuario ya que permite realizar varios juegos consecutivos con el mismo usuario, y gana el mejor de 3 partidas, esta modalidad también se basa en el juego “clásico” por lo cual todos los ataques y funciones del juego normal son iguales.

Roles

En cuanto al diagrama UML que se realizó, se pudieron identificar diferentes roles en las clases. Identificar los roles que hay en la solución provista es muy importante ya que ayuda a identificar el propósito de cada clase o darle una importancia no menor a cada una de estas.

Por otro lado también es útil identificar cada clase por su rol ya que se puede observar de una forma más clara que responsabilidad tiene cada clase, que en algunas ocasiones es bastante complicado dar una responsabilidad a cada clase sin antes haber definido su propósito.

Cabe destacar que las clases no pueden tener más de un rol, y todas deberían tener uno, ya que de esta forma nos ayuda a implementar nuestra clase a través de su finalidad.

A medida que se desarrolle el proyecto existe la posibilidad de que se añadan nuevas clases junto a sus roles, pero en el estado temprano actual del mismo identificamos los siguientes roles:

-Information Holder asignamos a las clases: User, CurrentGames, Ship & Statistics ya que su funcionalidad es almacenar y proveer información.

-Structurer se le asigna a la clase Ranking que vincula distintos objetos y almacena información sobre dicho vínculo.

-Service Provider la clase Board, ya que realiza trabajo y ofrece información y servicios a otra clase.

-Coordinator a las clases QuickChat y Commands cuyas funciones son de reaccionar y delegar tareas a otras clases.

-Controller las clases Bot y Game que se encargan de tomar decisiones y dirigir otras clases.

-Interfacer las clases Menu, Bomb, TimeTrial y Challenge ya que son clases que cambian información entre clases alterando el funcionamiento del juego principal.