

PRÁCTICA: HUNDIR LA FLOTA

Ingeniería del Software

Koldo Pérez de San Román
Denis Ionut Stefanescu
Borja Martínez Tadeo

Índice

- INTRODUCCIÓN	PÁG. 2
- ORGANIZACIÓN Y DESARROLLO	PÁG. 3
- DISEÑO COMPLETO	PÁG. 4
- CONCLUSIONES	PÁG. 5
- ANEXO: PRUEBAS UNITARIAS	PÁG. 6

Introducción

Este proyecto consiste en implementar una variante del tradicional juego “Hundir la flota”. Su dinámica básica es similar a la del juego de siempre. Tanto el jugador como el ordenador disponen de una flota compuesta por un portaaviones, dos submarinos, tres destructores y cuatro fragatas. El jugador dispone de dos tableros: el de la izquierda (donde colocará sus barcos) y el de la derecha (donde disparará para tratar de hundir los barcos del ordenador). En este proyecto, se dispondrá de dos tipos de armas: bombas (que destruyen la posición del barco sobre la que caen) y misiles (que destruyen por completo el barco). Además se dispondrá de herramientas complementarias como: radar (para ver si hay algún barco enemigo cerca de la posición en la que lo pongamos), escudo (para proteger un barco del impacto de una bomba o misil) y kits de reparación (para reparar barcos). Todo esto se podrá comprar en la tienda virtual del juego. El jugador empezará con una cantidad predeterminada de dinero. Con cada disparo recibirá dinero y en caso de darle a un barco recibirá el doble. El jugador y el ordenador dispondrán de turnos para realizar cada operación. Cuando acabe la partida se guardará la puntuación y el nombre del jugador en el fichero de puntuaciones.

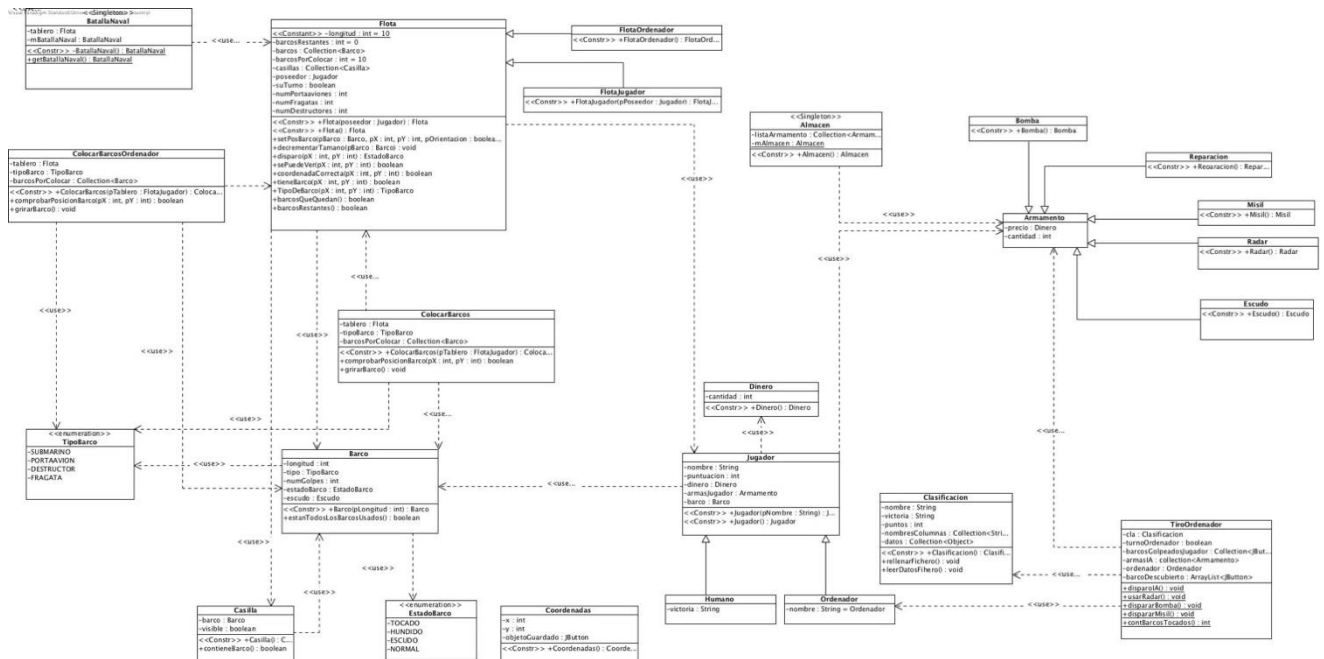
Organización y desarrollo

Para realizar esta práctica, no hemos tenido mucho en cuenta las historias de usuario para repartir el trabajo. Comenzamos por hacer el diagrama de clases en clase y a partir de ahí fuimos creando las clases en eclipse.

- **SPRINT 1:** Entre los tres empezamos a crear las clases y los paquetes para organizar el proyecto. Empezamos por la primera historia, la de crear el tablero y colocar los barcos. Denis y Borja fueron los encargados de crear las clases necesarias con sus constructores y parte de la interfaz gráfica. Koldo se encargó de implementar algunos métodos necesarios. También, se utilizó para dividir las tareas de los siguientes *Sprints*, establecer unos horarios de edición en *Eclipse* y así todos tener unos horarios preestablecidos y no entorpecer el avance del proyecto.

- **SPRINT 2:** En este sprint Borja se encargó de crear las ventanas de juego, tienda, puntuaciones, créditos y ayuda y darles funcionalidad a los botones. Denis se encargó de darle funcionalidad a la tienda (historia 10) y de crear los tableros en la ventana de juego. Koldo implementó más métodos necesarios para poder colocar los barcos. Entre Borja y Koldo sustituyeron los nombres de los botones de las armas por iconos. Denis se encargó de que el usuario pueda colocar los barcos cumpliendo los requisitos del enunciado y de que el ordenador coloque sus barcos después del usuario. Koldo creó una clase para poder poner una imagen de fondo en las ventanas y terminó la ventana de créditos. Denis realizó las historias 2, 3, 4, 5, 6 y 7: para disparar, usar los escudos y usar el radar. Por último, Borja consiguió hacer que las puntuaciones se guarden en un fichero y Denis finalizó el sprint implementando los test unitarios para las clases más importantes del proyecto.

- **SPRINT 3:** Como prácticamente todo el desarrollo de la práctica se ha realizado en los primeros dos *sprints*, el tercero se ha reservado para el desarrollo de la documentación y otros pequeños detalles de la implementación. También nos hemos encargado de completar y finalizar el diseño en UML. Por otra parte, destacar que este último *Sprint* lo utilizamos sobre todo para la implementación de las pruebas unitarias del proyecto.



Conclusiones

Tras la realización de este proyecto, podemos sacar diferentes conclusiones de este. La primera y la más importante, es la importancia del trabajo en equipo a través de la modalidad SCRUM. La cual nos ha servido saber llevar este proyecto en diferentes fases donde cada uno a elegido las tareas que desea hacer para llevar a cabo las diferentes historias de usuario.

Resaltamos la importancia y la ayuda que nos ha proporcionado la herramienta *Visual Paradigm* a la hora de realizar los diseños, ya que después esta herramienta nos permitía generar código en Java. Esto, nos hace evitar que tengamos menos errores cuando tengamos que relacionar clases...

Para la resolución de errores comunes con la GUI o a la hora de cuando diseñamos la base de datos, utilizamos el foro de *StackOverflow* el cual nos ayudó a la hora de crear los dos tableros de los barcos, comunicar a la base de datos para guardar al usuario o también para como volver a implementar la función *getResources()* para poder implementar los botones con iconos.

Como conclusión final, es destacar la importancia en grupo en este tipo de proyectos, así como la cooperación entre todos los miembros del grupo. Cuando a un miembro del grupo se le atascaba una tarea otro miembro le podía ayudar a cómo resolver sus errores. También el método de trabajo en 3 *Sprints* para la distribución de todo el proyecto, esto hace que todo el proyecto se pueda dividir y saber que se tiene que tener el proyecto 2-3 semanas antes de la entrega y utilizar el ultimo Sprint para solucionar errores, creación de test unitarios y completar el diseño y la documentación.

Bibliografía

- **Apuntes de la asignatura colgados en eGela.**
- **Para dudas en general:**
<http://www.stackoverflow.com>
<https://www.youtube.com>
- **Para poner imágenes de fondo:**
<http://www.driverlandia.com/java-anadir-background-a-jpanel/>