Metodología de la Programación

Grado en Ingeniería Informática

Curso 2024/2025

Villegas Rubio, Cristian
Silva Bienvenido, Raúl
Periñán Dávila, Alberto

Proyecto Modularidad

Hundir la Flota

Índice general

1. Introducción

1.1. Distribución de trabajo

El trabajo ha sido distribuido en distintos módulos, los cuales nos permiten amenizar la carga de trabajo individual, además de poder desarrollar nuestras habilidades individuales a la hora de desempeñar una actividad grupal relacionada con el desarrollo de un código amplio.

El proyecto consiste en los siguientes módulos:

- Configuración.
- Juego.
- Tablero.
- Barcos.

(En el punto 3.2 se desarrolla toda la información en referencia a los módulos del proyecto).

1.2. Organización de la documentación

La documentación nos sirve para poder conocer con mejor detalle y de una manera más natural la realización del proyecto.

En este documento diferenciamos cuatro partes (junto a las referencias externas utilizadas) : introducción, documentación de usuario, documentación de sistema y documentación de código.

Podríamos decir que los puntos más relevantes son las documentaciones de usuario y de sistema.

La documentación de usuario está orientada a aquellas personas que quieran conocer sobre el código, teniendo siempre un enfoque simple para que cualquiera pueda entenderlo con claridad.

La documentación de sistema es algo más técnica que la de usuario. En esta se desglosa de manera más detallada sobre el código en sí, como la implementación, el diseño, análisis... Junto a esto, aparece la definición detallada de los módulos, junto a las pruebas realizadas.

2. Documentación de usuario

A continuación nos encontramos con la documentación relacionada con los usuarios. Con esta documentación, el usuario podrá consultar todo lo relativo al programa, lo que este ofrece y como utilizar el programa para sacarle el máximo partido posible.

2.1. Descripción funcional

Nuestro programa consiste en la implementación simplificada del juego popular "Hundir la Flota". En este programa podremos realizar las acciones comunes del juego como disparar a unas casillas en concreto, colocar los barcos en las posiciones que el usuario desee, tocar y hundir barcos... Junto a estas funciones, el programa consta de algunos añadidos y restricciones que moldean el juego, entre estas, se encuentra lo siguiente:

- Los jugadores eligen el tamaño del tablero de juego que ambos deseen.
- Los jugadores eligen el tamaño máximo de los barcos y el número de estos, es decir, puede haber varios barcos del mismo tamaño.
- El tablero debe de tener el tamaño mínimo para que se puedan colocar todos los barcos elegidos por los jugadores en este.
- Los jugadores no podrán colocar un barco junto a otro, es decir, debe de haber como mínimo una separación de una casilla entre barcos para poder colocarlos en el tablero (esto también debe de tenerse en cuenta a la hora de la elección del tamaño del tablero).

2.2. Tecnología

Para este proyecto hemos utilizado una serie de tecnologías las cuales nos han ayudado a la hora de desarrollarlo. Aquí se incluye un listado de las principales tecnologías que hemos usado:

- **Visual Studio Code.** *VS Code* ha sido el editor de código fuente que hemos utilizado durante la realización del proyecto, esto debido a su flexibilidad y facilidad de uso, junto a la posibilidad de poder trabajar sobre un mismo código múltiples personas, agilizando el proceso de desarrollo.
- **Librería time.h.** Hemos utilizado la librería *time.h* que nos proporciona C para obtener la hora y fecha real del sistema, esta nos es de

ayuda a la hora de poder crear (como se verá en el módulo de juego) variables aleatorias.

- **Github Desktop.** La aplicación de sobremesa de *Github* nos ha sido de gran utilidad, ya no solo para comenzar a mostrar avances en nuestro perfil (el cual puede ser de gran utilidad para el futuro), sino que nos ha servido para poder compartir entre todos el proyecto, permitiéndonos así estar constantemente actualizados y dando acceso al trabajo de los demás de forma rápida y sencilla. Además de tener soporte con *VS Code*, lo cual hace que sea aún más cómodo el trabajar con esta herramienta.

- **Overleaf.** *Overleaf* es una página web en la que hemos realizado la documentación gracias a su facilidad de uso y soporte con LaTeX.

2.3. Manual de instalación

La instalación del programa es muy sencilla, lo único que hay que hacer es descargar los archivos del proyecto y ejecutar el archivo *.exe*. Al ser un programa que cuenta con una interfaz en consola no requiere de unas especificaciones hardware muy altas, por lo que no requiere prácticamente requisitos, sólo los mínimos para que el ordenador funcione con normalidad.

2.4. Acceso al sistema

Una vez lanzado el programa, la inicialización es muy sencilla, por no decir que es directa. Los jugadores deben de configurar los parámetros del juego y tras esto puede iniciar el juego con normalidad (hay que indicar que si ya se encuentra un juego guardado, los jugadores pueden elegir cargar la partida y esta se iniciará como la última vez que se jugó).

Si se desea salir del juego, los jugadores tienen varias formas de hacerlo. En el caso que quieran salir del juego en el menú principal, pueden elegir la opción *Salir* y saldrán del programa. Si se ha comenzado una partida y se desea salir, cada vez que un jugador dispare se mostrará junto a la información del jugador la opción *Guardar partida*, la que hará, como indica su nombre, guardar la partida y salir del juego.

2.5. Manual de referencia

En este apartado, se explica de forma más extensa las distintas funciones que puede realizar un jugador.

Configuración

Al iniciar el juego, el jugador debe seleccionar la opción *Configuración*, en la cual tiene las distintas opciones:

Configuración
1. Introducir datos
2. Mostrar
3. Borrar
4. Guardar
5. Cargar
6. Volver

- **1.Introducir datos.** En este apartado, el jugador debe introducir todos los datos personalizables relacionados con el juego, estos son: nombre del jugador, tipo de disparo (automático o manual), tamaño del tablero, tipo de barcos que desea utilizar, número de unidades de cada barco y el jugador que comienza la partida (elegido por consenso o por el sistema).
- **2. Mostrar.** Esta función se encarga de mostrar por pantalla un resumen de la configuración del juego, incluyendo a los jugadores.
- **3. Borrar.** Elimina la configuración creada.
- **4. Guardar.** Al elegir esta opción, se guardarán los datos de la partida en el fichero *Juego.txt*.
- **5. Cargar.** Relacionado con *Guardar*, se cargan los datos del fichero *Juego.txt*.
- **6. Volver.** Volver al menú principal.

Jugar partida

Una vez realizada la configuración, el jugador continúa eligiendo *Jugar partida*, con el que se mostrará un menú con las siguientes opciones:

Jugar partida
1. Jugar partida
2. Reiniciar partida
3. Resumen
4. Volver

- **1. Jugar partida.** Al elegir esta opción, se le muestra una bienvenida a los jugadores y el sistema pregunta a cada uno de ellos si prefieren colocar los barcos de forma manual o automática. Tras haber hecho la elección, comienza la partida con el jugador elegido, si este ha elegido disparar de forma manual, el sistema pregunta por las coordenadas que desea elegir el jugador, si no hay nada marcado en su tablero *oponente*, el sistema registra el disparo y muestra si este ha dado en el agua o en un barco enemigo. Tras cada disparo, el sistema pregunta si el jugador desea guardar la partida para asegurar que esta se conserve en caso de errores. Si se ha acertado el disparo, el sistema sigue preguntando por coordenadas al jugador hasta que este acabe dando en agua. Una vez fallado el tiro, se vuelve a preguntar si se desea guardar la partida, continuar sin guardar o guardar y salir de la partida. Si se elige continuar, ya sea guardando o sin guardar, se pasa el turno al contrincante, mostrándole las mismas opciones que al otro jugador, y continúa la partida hasta que se decida parar o uno de los dos gane la partida. Al ganar la partida, se mostrará un resumen de como ha quedado la partida.

- **2. Reiniciar partida.** Si los jugadores han terminado una partida, o desean que quieren empezar una partida de cero, pero desean mantener la misma configuración de la partida, al elegir esta opción se reinician todos los datos de la partida anterior para que los jugadores sólo tengan que colocar los barcos tras iniciar una partida, manteniendo su configuración.

- **3. Resumen.** Muestra por pantalla un resumen de la partida que se está jugando, mostrando datos como casillas hundidas, tocadas, barcos hundidos, restantes... A la vez que también muestra los tableros *flota* y *oponente* de ambos jugadores.

- **4. Volver.** Vuelve al menú principal.

Una de las ventajas más característicos del juego son las funciones automáticas. Estas se encargarán del disparo y del posicionamiento de los barcos, los cuales como se indica, se hará de forma automática si así lo desea el jugador. Gracias a estas funciones, los jugadores pueden tener una partida más dinámica y pueden poner a prueba el funcionamiento del programa. Estas opciones las deben elegir ambos jugadores y se pueden realizar distintas combinaciones, es decir, uno de los jugadores puede tener ambas opciones automáticas mientras que el otro realiza todo manualmente, o que un jugador decide que el disparo es manual pero la disposición de los barcos es automática...

3. Documentación de sistema

En esta sección nos encontramos con la documentación del sistema, la cual está orientada a un carácter más técnico que la de usuario. Aquí queda recogida la metodología utilizada para el trabajo y la descomposición en módulos del problema de forma más detallada, junto al plan de pruebas.

3.1. Especificación del sistema

Al tratarse de un proyecto con un tiempo de entrega limitado hemos seguido un desarrollo fluido y basado en la metodología de cascada, es decir, ir avanzando de fase en fase sin comenzar una nueva hasta no haber terminado la anterior.

En cuanto a la descomposición del problema, fue el primer aspecto que discutimos en cuanto se nos fue presentado el problema. Vimos que podíamos separar el problema inicial en subproblemas más fáciles de manejar debido al carácter del proyecto, por lo que realizamos la separación de estos subproblemas en varios módulos y repartiendo estos entre los distintos miembros del grupo para conseguir una mayor eficiencia a la hora de abordar el trabajo. Para esto, decidimos separar los módulos en: Configuración, Juego, Barcos y Tablero. Con esto, aseguramos que cada uno tuviese una carga de trabajo medianamente equitativa y repartir el problema entre todos, cumpliendo con el carácter modular.

3.2. Módulos

En esta sección se muestran los distintos módulos que se han utilizado para el proyecto, mostrando una descripción de estos y sus funcionalidades.

Configuración

Encontramos todo lo relacionado a, como dice su nombre, la configuración del juego. En este se encuentra la estructura principal, la cual almacenará todos los datos necesarios para que los demás módulos puedan manejarlos de forma eficiente.

El módulo cuenta con distintas funciones, las cuales se encargan de cada uno de las opciones que se muestran en el menú de configuración como *introducirDatos, mostrarDatos, guardarDatos* y *cargarDatos*. Además de estas funciones, tenemos *menuConfiguracion*, que muestra por pantalla al jugador estas opciones, tras esto, devuelve un valor que nos servirá para saber si se

ha cargado o no una partida. También existe la función *menuBarcos* que lleva al menú donde podremos crear barcos nuevos y modificar, eliminar y listar los existentes. Como función auxiliar tenemos *comprobarTamano*, esta se encarga de comprobar que el tamaño elegido para los tableros permite que se puedan colocar de forma correcta los barcos. Y por último, contamos con *barcosHundidos*, esta función nos sirve para que, si se carga una partida, se sepa cuales han sido los barcos que se han hundido en esta.

Juego

Aquí se encuentra el "núcleo" del juego en sí. Encontramos todas las opciones relacionadas con el menú de *Jugar partida* y algunas funciones auxiliares que sirven para hacer más asequible el código. Todo lo relacionado con las funciones principales del módulo se encuentra en el punto *2.5*, en el que se explica con detalle lo que hace el sistema.

En cuanto a funciones auxiliares, nos encontramos las siguientes:

- **MostrarTableros**. Esta función se encarga, como dice su nombre, de mostrar los tableros del jugador elegido. Se decidió hacerla de manera auxiliar ya que esta función se utiliza en distintos puntos del código, por lo que es más cómodo hacer esto.

- **disparoAutomatico**. En caso de que el usuario decida que el disparo se va a realizar de forma automática, esta función se encarga de hacer esto mismo, el sistema es el que decide donde realizar el disparo. Al principio, se elige al azar y si, el sistema acierta, elige el sentido que quiere tomar para poder disparar, si falla, cuando llegue su turno de nuevo, probará con otro sentido y así hasta dar con la posición acertada, tras esto seguirá esa dirección hasta hundir el barco y empezar de nuevo el proceso.

- **comprobarDisparo**. Esta función es una de las funciones auxiliares más importantes, pues esta es la encargada de comprobar si un barco ha sido hundido o no. La comprobación la realiza comparando entre el tablero *flota* del enemigo y *oponente* del que realiza el disparo, comprobando todas las direcciones posibles que puede tener el barco, recorriéndolo para conocer su tamaño, inicio y fin, y comparando si en el tablero *oponente* están marcadas todas las casillas que se ocupan en el tablero *flota* del enemigo.

- **primerDisparo**. Esta función es auxiliar de la función *disparoAutomatico*, pues esta sirve para elegir de forma aleatoria el primer disparo

del sistema. Al igual que *MostrarTableros*, se decidió hacerla de manera auxiliar por comodidad, debido a su aparición en varios puntos del código.

Tablero

Este módulo es el encargado con todo lo relacionado con el uso y manejo de los tableros. Cuenta con las siguientes funciones:

- **generarTablero**. La más importante, pues sin ella no podría realizarse el juego. Encargada, como dice su nombre, de generar los tableros de los jugadores que serán la base de todo.
- **asignacionManual**. En caso de que el jugador decida colocar sus barcos de forma manual, esta función se encarga de guiar a este de la manera más eficiente y correcta. También avisará al usuario en los casos que no pueda colocar un barco en la posición que desea, ya sea porque no sea una posición válida o porque no se le permite.
- **asignacionAutomatica**. En el caso de que el jugador prefiera que los barcos se coloquen de forma automática, esta función hace que el sistema sea el encargado de colocar los barcos en el tablero *flota* de forma correcta, siguiendo las normas establecidas por el programa.

Barcos

En este módulo, nos centramos más en el manejo de los barcos del juego y todo lo relacionado con estos.

Encontramos las siguientes funciones en este módulo:

- **menuBarcos**. Esta función es la encargada de mostrar al jugador las distintas opciones que puede realizar con los barcos (esta función se mostrará en *Configuración*).
- **cargarBarcos**. Carga los barcos existentes en un fichero donde se guardan los barcos.
- **mostrarBarcos**. Al elegir esta opción, se muestra por pantalla todos los barcos existentes en el fichero.
- **crearBarco**. El jugador decide crear un barco nuevo en el que decide su nombre y tamaño.

- **eliminarBarco**. Si el jugador desea eliminar un barco existente en el fichero usará esta función para su eliminación.
- **modificarBarco**. Parecida a *crearBarco*, solo que sirve en caso de que el jugador desee modificar un barco ya existente, pudiendo elegir tanto el nombre como el tamaño, al igual que a la hora de crear uno.
- **obtenerNBarcos**. Función auxiliar que nos sirve para saber el número de barcos existentes en el fichero.
- **obtenerIDBarco**. Función auxiliar que nos sirve para conocer el ID de un barco en concreto.
- **guardarBarcos**. Si el jugador decide crear o modificar un barco, automáticamente se guardará el cambio en el fichero llamando a esta función.
- **leerCadena**. Función auxiliar que nos sirve para leer el nombre de los barcos.

3.3. Plan de prueba

En este punto encontramos en detalle el plan de prueba que hemos seguido para comprobar que el código funciona correctamente. Distinguimos pruebas de los módulos y de integración, junto al plan de pruebas de aceptación.

Prueba de los módulos

Las pruebas de los módulos nos sirven para saber que cada función y cada módulo funcionan correctamente de forma individual. Es importante tener en cuenta el carácter unitario de estas pruebas, pues no sirve de nada el tener unificado el proyecto cuando los módulos y sus funciones no funcionan independientemente.

Debido a esto, decidimos al inicio del proyecto que estas pruebas eran las más importantes a la hora de realizar nuestros módulos y le dimos la prioridad que las caracterizan.

Para realizar estas pruebas, nos basamos en el principio de las tres A's (*Arrange, Act* y *Assert*). Este principio consiste en seguir los siguientes pasos: definir los requisitos que debe seguir el código, ejecutar la prueba y obtener sus resultados. Tras obtener estos resultados, comprobamos si estos son los esperados, si no es así, se corrige el error y se repite el proceso.

Junto a estas pruebas, también se han realizado pruebas de ruta básica. Estas pruebas estarán adjuntas en un fichero *.pdf* con un ejemplo de este tipo de prueba realizado por cada uno de los integrantes.

Prueba de integración

Una vez realizadas todas las pruebas de los módulos necesarias, hay que comprobar el funcionamiento y compatibilidad de los módulos entre sí.

Primero, comenzamos comprobando la compatibilidad entre los módulos necesarios para que el programa pueda funcionar, es decir, que sin estos módulos no sería posible iniciar el programa, en este caso, estos módulos son los de configuración, barco y el propio main, pues sin estos, el programa no se podría ni iniciar ni configurar. Para comprobar su correcto funcionamiento, realizamos una prueba como si un usuario estuviese iniciando una nueva partida, configurando sus datos y los de otro jugador.

Tras haber comprobado que funciona correctamente, añadimos el resto de módulos y vamos comprobando una por una las funcionalidades de estos, viendo que los resultados son los esperados.

Plan de pruebas de aceptación

Las pruebas de aceptación no las podemos realizar los integrantes del proyecto, pues estas son realizadas por el usuario del sistema que prueba el resultado en su propia computadora. Aún así, especificamos algunas pautas que puede seguir el usuario con las que nos aseguramos de que el sistema funciona correctamente. Para ello, el usuario puede basarse en toda la información que se encuentra en el punto *2.5*, el cual contiene todo lo básico que debe conocer el usuario para ver el funcionamiento del sistema sin entrar en mucho detalle de como este realiza esas funciones. Si el usuario decide no consultar este punto, o simplemente decide probar el programa a su manera deberá reportar los fallos que encuentre para que el equipo pueda localizarlo y solucionarlo, ya que puede que este se haya escapado en la fase de pruebas.

Referencias

[1] Stack Overflow. (2025). Stack Overflow - Where Developers Learn, Share, & Build Careers. Stack Overflow. https://stackoverflow.com/

[2] GitHub. (2025). Build and ship software on a single, collaborative platform. GitHub. https://github.com/

[3] Overleaf. (2025). Overleaf User Manual. Overleaf. https://www.overleaf.com/learn