

AWSttle

A BATTLE FOR THE CLASSROOM

INFORME TÉCNICO

FECHA LANZAMIENTO

21/05/2021



AIDA IGLESIAS ALCALÁ
CARLOS VALENZUELA GARCIA
SANDRA BEL GIL



- ★ FANTASÍA
- ★ RPG POR TURNOS
- ★ MEDIEVAL
- ★ CREATIVO

EMPEZAR JUEGO



ÍNDICE



Resumen del proyecto.....	3
Tecnologías aplicadas.....	5
Incidencias técnicas.....	6
Soluciones.....	9
Propuestas de mejoras.....	11
Valoración personal.....	12

RESUMEN DEL PROYECTO



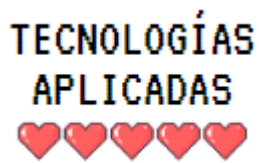
En el proyecto hay que llevar a cabo la creación de un juego el cual consiste en simular batallas entre personajes con distintas razas: humanos, elfos y enanos. Estos tienen características las cuales varían según el tipo de raza que escoges: puntos de vida, agilidad, velocidad, defensa, fuerza y arma; Para escoger el arma debes tener en cuenta la raza, ya que no todas pueden portar las mismas armas.

La mecánica del juego es similar a un videojuego de rol por turnos. Una vez tenemos a nuestro personaje con su arma seleccionada, se genera un enemigo al azar, seguido comienza la batalla. Ataca primero el personaje con mayor velocidad, en caso de empate ataca en primer lugar el personaje con mayor agilidad, si esta coincide de nuevo se elige al azar el personaje que debe realizar el primer ataque. El combate finaliza cuando uno de los dos personajes se queda sin puntos de vida. Al finalizar se muestra el resumen del combate y nos da la opción de volver a jugar o salir del juego. También, podemos ver el ránking de los jugadores que han realizado mejores partidas, desde el mejor jugador hasta el décimo.

Para poder realizar el proyecto correctamente, debemos aplicar los conocimientos aprendidos con anterioridad en los diferentes módulos formativos que formaban parte del proyecto. En cada módulo se nos pide diferentes requisitos:

- **M1.- Sistemas informáticos**
 - Manual para el usuario
 - Informe técnico
 - Diagrama de Gantt
 - Power Point para la presentación
- **M2.- Base de datos**
 - Creación de la BBDD
- **M3.- Programación**
 - Mecánica del juego
 - Interfaz gráfica


- **M5.- Entornos de desarrollo**
 - Gestión de archivos mediante GitHub
 - Diagrama de actividades
 - Diagrama de casos de uso



Para la correcta realización del proyecto hemos utilizado diferentes tecnologías, desde editores de imágenes hasta aplicaciones para el entorno de desarrollo.

- **IntelliJ y Eclipse (Java)**
 - Creación del programa
 - Interfaces gráficas
- **MySQL Workbench**
 - Creación de BBDD
 - Inserción de datos
- **Photoshop**
 - Creación de iconos
 - Edición de imágenes
 - Creación de portada
- **GitHub**
 - Controlador de versiones
 - Gestión de archivos
- **Sistemas operativos**
 - Windows
 - Ubuntu

INCIDENCIAS TÉCNICAS



- **Problemas con diferentes Sistemas Operativos y programas**

La realización del proyecto se ha dado desde dos sistemas operativos diferentes, Windows y Ubuntu. También, se ha trabajado en él desde dos aplicaciones de entornos de desarrollo diferentes, IntelliJ y Eclipse.

En el inicio del proyecto cada integrante de este realizaba una parte del trabajo y estas se subían a GitHub para poder trabajar siempre con el código más actualizado posible. En el momento de importar dicho código a nuestros ordenadores, se generaban diversas incidencias y errores causadas por la utilización de diferentes sistemas operativos y programas (proyecto descuadrado, desorganización de carpetas, código no printa el resultado esperado, entre otros).

Esta incidencia no solo se ha dado trabajando desde clase, también ha ocurrido trabajando desde casa, ya que todos utilizamos diferentes sistemas operativos y aplicaciones en nuestros ordenadores principales.

- **Descuadre de interfaz gráfica en distintos Sistemas Operativos**

El hecho de trabajar en diferentes Sistemas Operativos causó el descuadre de las interfaces gráficas de diferentes clases, como en la clase Ranking o clase EndFight. Ambas clases se trabajaron en Ubuntu y se descuadraron completamente al importarlas a Windows.

- **Desaparición de imágenes en interfaz gráfica**

Al inicio del proyecto, no aparecía ningún tipo de imagen. Nosotros trabajamos con las fotografías de los personajes y los fondos de las ventanas y estas no se mostraban.

- **Tipografías incompatibles con distintos Sistemas Operativos**

Al añadir distintas fuentes de texto a nuestro proyecto para mejorar la apariencia visual de este vimos que no se aplicaba ningún cambio.

El código introducido para realizar la modificación de la fuente del texto estaba correctamente, no daba ningún error de ejecución ni compilación. Por ello, decidimos probar en otro programa de entorno de desarrollo para intentar evitar esta incidencia. Una vez introducido el código en el nuevo programa seguía sin mostrar el resultado esperado, de esta manera llegamos a la conclusión de que no todas las fuentes eran compatibles con nuestro Sistema Operativo (Ubuntu). Para confirmar nuestra teoría, probamos el mismo código en Windows y sí que mostraba el resultado esperado.

Buscamos por internet un método que nos mostrase por pantalla las fuentes disponibles en nuestro sistema operativo, muchas de las fuentes no eran compatibles con Ubuntu y muchas otras no lo eran con Windows y por ello no se realizaba ningún tipo de cambio.

- **Descuadre de las imágenes de los personajes**

En nuestra clase Character se encontraban cargados los datos de la base de datos desde los cuales podíamos acceder tanto a las fotografías de cada uno de los personajes como a sus descripciones. Esta clase se creó en Windows y cuando quisimos trabajar con ella desde Ubuntu la imagen y descripción de uno de los personajes (en este caso, Elfo) estaba descuadrada a diferencia de las demás. Modificamos ciertos parámetros del código para ver cómo actuaba; este mostraba el resultado esperado, por ende, el problema no era del código ni la aplicación de entornos de desarrollo.

Revisando la base de datos del videojuego observamos que la descripción de ese personaje ocupaba más espacio que la descripción del resto de personajes de su misma raza. Conseguimos dar con la incidencia pero, esta solo se daba en Ubuntu, ya que en Windows no cambiaba el formato aunque la descripción de uno de los personajes fuese más extensa que la del resto.

- **Alineación de los paneles**

A la hora de maquetar las interfaces gráficas decidimos añadir los diferentes componentes (JButton, Image Icon, JLabel....) dentro de distintos JPanels. Con los días nos percatamos que estábamos trabajando con un método mucho más confuso y trabajoso, ya que muchos paneles no se colocaban en el lugar esperado, se descuadraban, teníamos paneles sueltos y perdíamos mucho tiempo arreglando las ventanas.

- **No-actualización de las pantallas**

En el momento de iniciar la partida los datos no se actualizaban correctamente en las diferentes pantallas. El ejemplo más notorio era con la clase Ranking: siempre salían los mismos jugadores a no ser que se reiniciase el programa.

SOLUCIONES



- **Problemas con diferentes Sistemas Operativos y programas**

La solución óptima para evitar el gran volumen de incidencias causadas por la utilización de diferentes sistemas operativos sería utilizar el mismo sistema operativo en todos los integrantes del grupo.

Por un lado, los ordenadores de nuestra aula tiene Ubuntu como sistema operativo principal, imposibilitando poder trabajar todos desde el mismo sistema operativo. Por otro, trabajar desde una Máquina Virtual Windows ralentizaría el trabajo.

Por ello, seguimos trabajando con Windows y/o Ubuntu.

- **Descuadre de interfaz gráfica en distintos Sistemas Operativos**

Volver a maquetar correctamente los diferentes descuadres visuales.

- **Desaparición de imágenes en interfaz gráfica**

Al insertar las fotografías en la base de datos con las extensiones .jpg y .png estaban en minúscula, mientras que en la carpeta del programa estaban en mayúscula.

Para solucionar la incidencia, cambiamos las extensiones a mayúsculas en la base de datos.

- **Tipografías incompatibles con distintos Sistemas Operativos**

Decidimos no añadir ninguna tipografía e intercambiar los textos de los botones por imágenes.

Con este cambio, evitamos errores de compatibilidad con las fuentes y a su vez conseguimos que nuestro programa se viese original.

- **Descuadre de las imágenes de los personajes**

Se redujo la descripción del personaje afectado a dos líneas para evitar errores en Ubuntu.

- **Alineación de los paneles**

Para maquetar correctamente, decidimos crear dentro del panel general otros paneles principales donde crearíamos la estructura base y dentro de ellos añadiríamos directamente los componentes. Al realizar la maquetación de esta manera, no solo el código queda más claro sino que es mucho más sencillo de realizar.

- **No-actualización de las pantallas**

Creamos un método que reinicia las pantallas cuando estas se han cargado completamente, así los datos siempre aparecen actualizados.

PROPUESTAS DE MEJORAS




A diferencia del primer proyecto realizado, ha habido una gran mejoría en la explicación de las diferentes entregas.

No obstante, se tendrían que haber facilitado las rúbricas del proyecto al inicio de este y no la última semana para facilitar la organización y realización del trabajo. La entrega tardía del documento con las diferentes rúbricas ha causado la modificación de ciertas partes del proyecto las cuales teníamos realizadas y no esperábamos modificar.

Por otro lado, el cambio de calendario escolar ha provocado que el plazo de entrega se adelante y nos hayamos tenido que organizar más rápido de lo planeado. De igual forma, el profesorado nos ha facilitado diferentes ayudas para no agobiarnos con el proyecto. Sin embargo, teníamos que entregar un gran volumen de prácticas simultáneamente a la realización de este.

Por ello, para futuros proyectos creemos que es conveniente que se siga manteniendo este método de explicación, ya que ha sido mucho más claro desde el inicio. También, facilitar las rúbricas necesarias para la correcta organización y realización del trabajo y de misma forma no modificar el calendario estipulado para que la organización del proyecto final sea más precisa.

VALORACIÓN PERSONAL



Gracias al proyecto final hemos aprendido a trabajar con personas con las que no estábamos acostumbrados. Por ello, hemos mejorado la planificación y realización del trabajo en equipo. No obstante, tenemos mucho que mejorar en cuanto a nuestra organización y reparto del trabajo.

También, hemos mejorado a la hora de trabajar y asimilado diferentes conceptos que desconocíamos. A raíz de realizar el proyecto, hemos practicado código Java, tanto interfaces gráficas cómo realización de un programa, base de datos en MySQL, gestión de archivos en GitHub y documentación del trabajo efectuado.

Por ello, creemos que la realización de proyectos ayuda a aprender de manera dinámica y entretenida diferentes aspectos que nos ayudarán en un futuro laboral. Por un lado, ponemos en práctica los conocimientos adquiridos en el módulo y por otro los conocimientos de crecimiento personal.