

ESPECIFICACIÓN DE REQUISITOS DEL SOFTWARE

Doomies

Proyecto Final

Programación, Entornos de Desarrollo y Bases de Datos

Néstor de
Frutos, Víctor
González,
Javier Mérida y
Lena Simón
IES Julián
Marías
Valladolid



**Junta de
Castilla y León**
Consejería de Educación

Consejería de Educación | Junta
de Castilla y León

Contenido

0. Versionado del documento	2
1. Objetivos generales del proyecto	2
Requisitos funcionales	3
Requisitos no funcionales.....	4
Documentación exigida	5
Alcance.....	5
2. Descripción del software	6
Resumen de funcionalidad.....	6
Herramientas técnicas	6
Diagramas básicos de funcionalidad	7
3. Valoración prevista	9
Estimación de esfuerzo	9
Estimación en fechas.....	9
4. Análisis de riesgos.....	10
5. Otros comentarios	10

0. Versionado del documento

Versión	Fecha del cambio	Cambios introducidos
0	05/04/2022	Confección inicial del documento (plantilla)
1	14/04/2022	Primera escritura del documento
2	19/04/2022	Modificación de los casos de uso, entregables, estimación de fechas y la historia del juego
3	20/04/2022	Modificación de los casos de uso y corrección de palabras
4	26/04/2022	Modificación de la sección Entregables (Documentación exigida)
5	02/06/2022	Modificación del diseño del documento y añadido tareas a realizar en un futuro y herramientas técnicas usadas.

1. Objetivos generales del proyecto

Se desea construir un juego basado en programación *Java*. Permitirá el desarrollo en tres aspectos:

- 1- Conocimiento: se utilizará todo el aprendizaje adquirido durante el curso, haciendo un repaso general de los contenidos asimilados en los distintos módulos.
- 2- Cohesión: se adquirirá una idea más clara de las interconexiones entre las distintas materias, viendo su relación en un proyecto global.
- 3- Trabajo en equipo: servirá de primer contacto con el ambiente empresarial en el que trabajarán los alumnos en un futuro:
 - a. Profesionalidad.
 - b. Esfuerzo común y entendimiento.
 - c. Gestión de roles.

Características del proyecto

- Juego con interfaz gráfica tipo *Scroller*.
- Diferentes tipos de enemigos.
- Basado en el juego *Doom*.
- El protagonista dispara con una pistola.
- Animaciones.
- Varios niveles jugables con diferente ambientación.

Requisitos funcionales

El juego ha de realizarse con el lenguaje de programación *Java*, siguiendo las siguientes premisas:

1- Dificultad dentro de un rango exigible:

- Idea innovadora
- Cohesión entre módulos
- Se requiere uso de herramientas vistas:
 - o Parte gráfica (*Awt* y *Graphics*).
 - o Uso de programación secuencial.
 - o Programación orientada a objetos.
 - o Conexión con Bases de Datos.
 - o Gestión de ficheros.
 - o Programación multihilo.

2- Foco en la ejecución del proyecto:

- Organización del equipo, con el uso de herramientas: *Teams*, *Trello*, *Github*, *Discord*, etc.
- Jugabilidad.
- Programación robusta (funcionamiento sin errores ni problemas de rendimiento).

Gestión de roles.

- o Análisis: Néstor de Frutos.
- o Diseño: Lena Simón.
- o Codificación: Víctor González.
- o Pruebas: Javier Mérida.

3- Presentación:

- Unos 35 minutos.
- Dinámica e interesante.
- Explicada claramente.

4- Requisitos:

Requisito	Descripción	Detalles
Requisito funcional 1	Nueva partida	El juego permitirá seleccionar un nivel.
Requisito funcional 2	Seleccionar nivel	Una vez seleccionado el nivel se buscará en el fichero correspondiente del mapa y permitirá jugar el nivel.
Requisito funcional 3	Guardar partida	El juego permitirá guardar partida en el fichero correspondiente.
Requisito funcional 4	Jugar nivel	El nivel deberá tener una lógica y se tendrá que visualizar el mapa, el funcionamiento del personaje, el arma, los enemigos, los movimientos y las interacciones entre seres.
Requisito funcional 5	Menú de pausa	Será necesario un menú que permita salir del juego o seguir jugando.
Requisito funcional 6	Cargar partida	Habrà una opción en el menú principal que almacene las partidas guardadas en ficheros con su nombre y fecha.
Requisito funcional 7	Salir	Será necesario un botón de salir para cerrar el juego.

Requisitos no funcionales

Requisito	Descripción	Detalles
Requisito no funcional 1	Ejecutable con JRE	Para poder jugar, el equipo debe tener un JRE instalado
Requisito no funcional 2	Versión 1.5	Se puede jugar en la versión de java 1.5 o superiores
Requisito no funcional 3	60 FPS	El equipo deberá poder cargar el juego a 60 FPS o superiores.

Documentación exigida

ENTREGABLES	
<input checked="" type="checkbox"/>	ERS - Documento de Especificación de Requisitos del Software
<input checked="" type="checkbox"/>	PP - Plan de pruebas inicial.
<input checked="" type="checkbox"/>	IP - Informe de pruebas.
<input checked="" type="checkbox"/>	DT - Diseño Técnico
<input type="checkbox"/>	DF - Diseño Funcional
<input checked="" type="checkbox"/>	MU - Manual de usuario
<input checked="" type="checkbox"/>	PI - Plan de implantación
<input type="checkbox"/>	ID - Índice de documentación

Alcance

Se realizará el proyecto según los requisitos acordados, la documentación a entregar es la descrita en el apartado anterior.

El juego no va a tener modo multijugador.

No se va a utilizar nada de HTML, CSS, XML en la realización del proyecto.

El juego no va a ser traducido a otros idiomas.

El juego no va a tener gráficos 3D.

- Tareas a realizar en un futuro

- Añadir nuevos mapas y nuevas temáticas de mapas.
- Añadir nuevos enemigos con nuevas funcionalidades.
- Añadir música y todo tipo de sonidos, como el sonido de los disparos o el de los enemigos al morir.
- Añadir un apartado de logros.
- Meter objetos por el mapa que tengan funcionalidades.
- Añadir nuevos personajes y tener la opción de elegir con cuál de ellos jugar.

2. Descripción del software

Vamos a construir un juego tipo *Scroller* basado en el *Doom* donde habrá que pasarse los niveles llenos de horripilantes enemigos y demonios.

Resumen de funcionalidad

Se tratará de un juego tipo *Scroller*.

El funcionamiento será el que sigue:

- Habrá un menú de inicio.
- Habrá una selección de nivel.
- Habrá que recorrer el nivel con diferentes enemigos, en el que el personaje podrá correr, saltar disparar a lo largo de un mapa.
- Se finalizará el nivel.
- Habrá una opción de salir del juego o de volver a jugar.
- Habrá una opción de guardar partida.
- Habrá una opción de cargar partida.

Herramientas técnicas

Se usará *Java* como lenguaje de programación:

- *Java* v.8.
- *JDK* v.18.
- *JRE* v.8.

Se desarrollará el juego en el entorno de desarrollo *Netbeans* v.13.0.

Se integrará como *Gui*:

- *Swing*.
- *Awt.Graphics*.

Se usará *XAMPP* v.3.3.0 para lograr una conexión con base de datos óptima.

Se usarán herramientas para la creación de los *sprites* del juego (bloques, personaje y enemigos):

- *Pixilart*.
- *Piskel* v.0.14.0.

Se usará *Gimp* v.2.10.12 para la modificación de las imágenes de fondo.

Se utilizará *Diagrams.net* para hacer los diagramas de la documentación.

Se utilizarán herramientas para ayudar a la codificación:

- *Git* v.2.36.1.
- *Github*.

Se utilizarán herramientas para la organización del grupo:

- *Teams.*
- *Trello.*
- *Discord.*

Se utilizarán herramientas de edición de documentos:

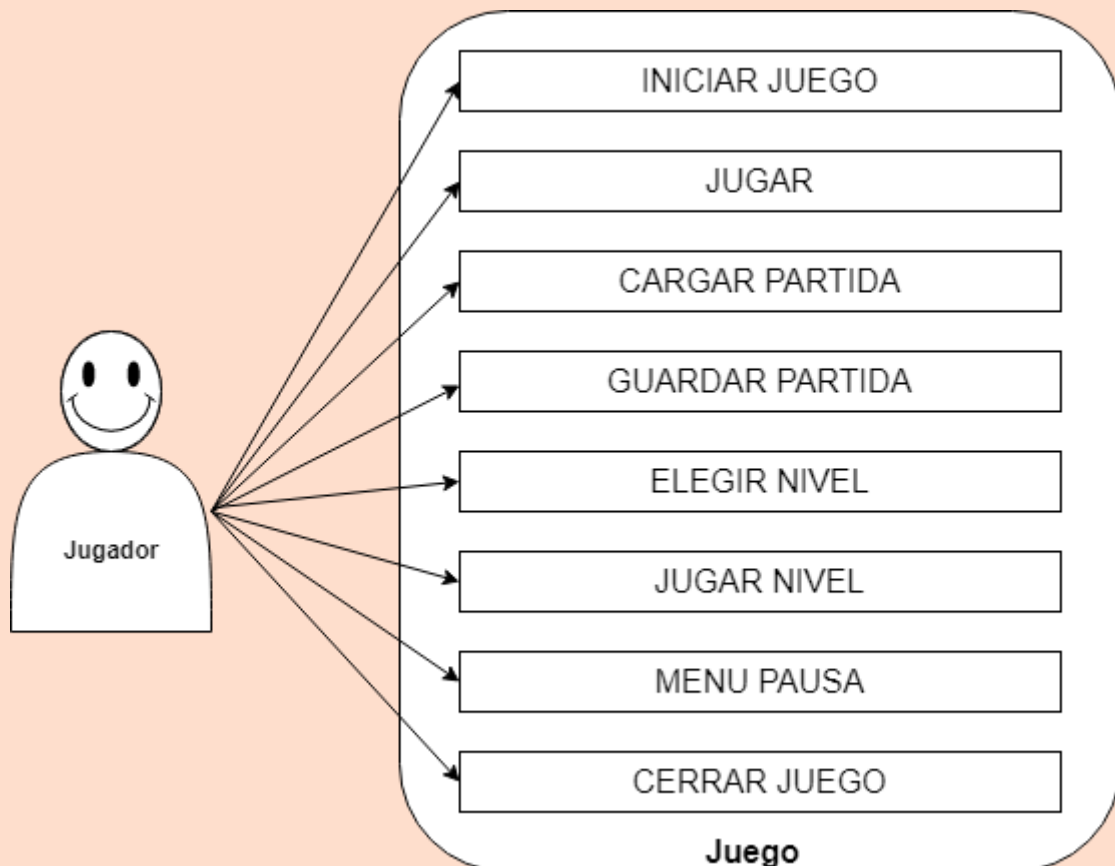
- *Excel.*
- *Word.*

Se usarán herramientas para la edición y grabación del manual de usuario:

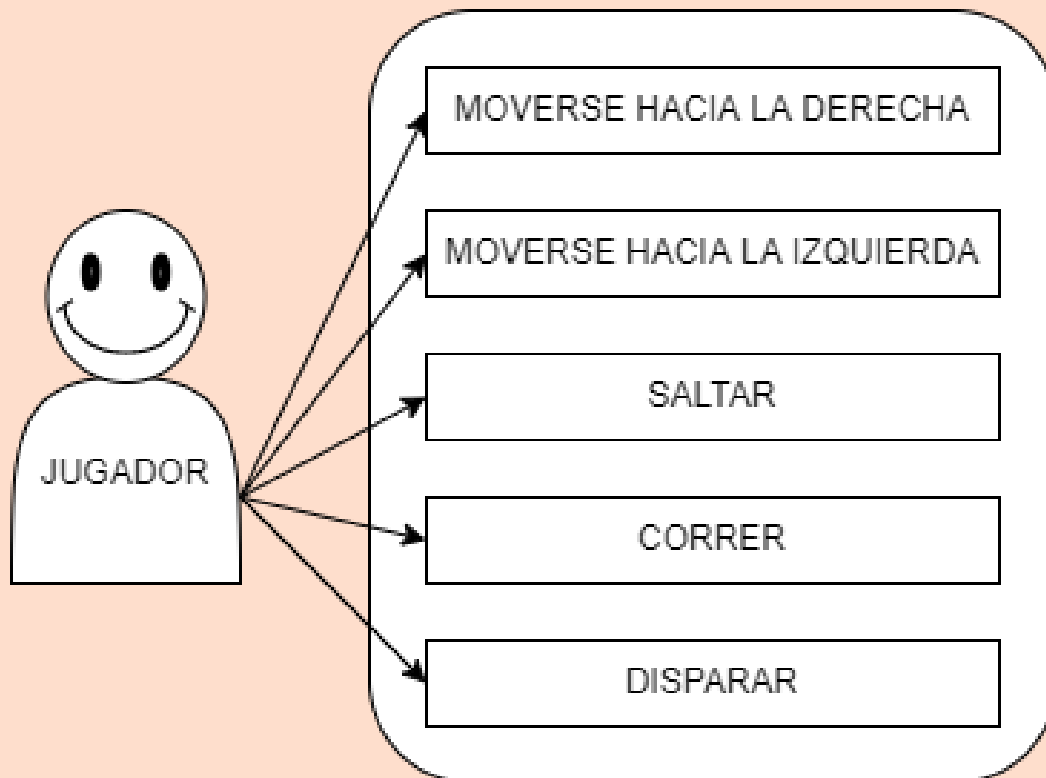
- *Openshot v.2.6.1.*
- *Obs v.27.2.4.*
- *Droid camera v.6.5.2.*

Diagramas básicos de funcionalidad

Funcionalidades del juego.



Funcionalidades del movimiento del personaje



3. Valoración prevista

Estimación de esfuerzo

Fase	Tarea	Horas
Análisis	Realización del análisis	15
	Documentación ERS	2
Diseño	Gestión de análisis	25
	Diagramas	10
	Documentación DT	3
	Diseño de pruebas: PP	20
	Gráficos	20
	Gestión de BBDD	50
Codificación	Codificación	150
Pruebas	Realización de pruebas: IP	120
"Mantenimiento"	Generación de vídeo: MU	9
	Preparación de presentación	9
TOTAL		433

Estimación en fechas

Hito	Fecha	Comentarios/Dependencias
Comienzo del proyecto	01/04/2022	Cuando se dieron los requerimientos.
Finalización de análisis	18/04/2022	Resolución de dudas del análisis.
Finalización de diseño	25/04/2022	Resolución de dudas del diseño.
Inicio de codificación	20/04/2022	Depende de la creación del UML.
Inicio de pruebas	27/04/2022	Su inicio depende de la codificación, se necesita tener un prototipo jugable para poder empezarlas.
Entrega definitiva	03/06/2022	Depende de los retrasos que puedan surgir a lo largo de todo el proyecto, habrá que intentar arrastrar el mínimo retraso posible.
Defensa del proyecto	06/06/2022	Pendiente verificación por parte de profesor.

4. Análisis de riesgos

Existen dependencias con el progreso de la asignatura de Programación: se exige el uso de integración con base de datos y ficheros, y todavía no se han dado los contenidos. Esto puede retrasar la gestión en esos puntos.

También dependemos de poder crear los *sprites* para antes de la fecha adecuada.

Existe el riesgo de que no podamos realizar las animaciones, o que tampoco consigamos programar las colisiones de las *hitbox* del juego.

Existe el riesgo de usar programación multihilos, por culpa de su complejidad y no haberla dado este año, podría retrasar el proyecto considerablemente.

Existe el riesgo de al intentar aspirar a un proyecto de alto nivel, nos compliquemos por el camino y surjan problemas impidiéndonos llegar a la meta deseada.

5. Otros comentarios

Historia del juego:

Una nave espacial ha entrado en órbita de un planeta desconocido, lo que ha provocado su colisión. El jugador tendrá que enfrentarse a horripilantes monstruos por distintos niveles.

El profesor de Entornos es demasiado friki.