

RESUMEN PRESUPUESTARIO

Introducción

En este documento se busca plasmar toda la información respectiva a los entregables que se establecieron en el presupuesto a principio del proyecto. Por ello se explicaría qué entregables están desarrollados, cuáles no y la razones que nos han llevado a esa decisión. Muchos de los entregables son obligatorios, así que pueden carecer de una explicación o de una más reducida.

Para hacerlo de forma organizada se categorizarán por sus respectivas asignaturas y si han sido realizadas o no. También hay mecánicas de juego u otras implementaciones que no son un entregable en sí, pero se añadirán comentarios en las asignaturas afectadas para explicar los cambios realizados.

Videojuegos I

Entregados	Explicaciones
Documento de diseño de mecánicas de los NPCs	Nos era necesario para establecer la programación gameplay del juego y crear futuros diseño en base al documento.
Documento de diseño de sistemas de toma de decisión	El documento marca el conjunto de decisiones que va tomar la IA y nos ayudó a establecer las bases de la misma.
Documento de diseño técnico de la arquitectura de la IA	Estructurar la inteligencia artificial y así conocer sus diferentes dimensiones y que funcionen tienen que cumplir.
Gestión de estados de la IA con Maquinas de Estados	Estructurar el sistema decisión de forma más simple para el rehén ya que no hacía falta una complejidad similar a los enemigos.
Sistema de toma de decisión con Behaviour Trees	Conocer el sistema de programación en árbol y así conseguir una estructura de toma decisión sencilla, pero con muy buena respuesta de decisión.
Sistema de búsqueda de caminos y control (Pathplanning/following)	Planificación y búsqueda de los caminos necesarios para desplazarse por el mapa.
Comunicación simulada básica entre NPCs (BlackBoard)	Nos proporcionaba la comunicación entre los objetos del mundo y los NPC para gestionar varias de sus pautas como comer, beber y/o curarse ...

Sistema de gestión de eventos (Trigger System/Event Manager)	Gestor necesario para la comunicación de eventos a los diferentes npcs como el ruido o la activación de la alarma.
Sistema de Waypoints para pathfinding continuo	Grafo que gestiona los puntos en el mapa creando una máscara por donde los npc pueden circular por el mapa.
Entrega modificada	
Sistema de memoria de estado y reacción.	Este sistema se encarga de gestionar la percepción de los enemigos para ver cambios en su entorno. Este sistema se halla incompleto y su funcionamiento es correcto, pero al igual que el sistema incompleto todo acarreado por la falta de tiempo.
Sistema de percepción sensorial (vista, oído, olfato, canales ...)	Nuestro sistema tiene sensores de visión y oído pero gestionados de forma diferente a la requerida, es decir, se han creado los sensores, pero no con la implementación deseada.
No entregado	
Sistema de depuración visual in-game de la IA	Muestra gráfica del funcionamiento de la IA. SE comenzó un proyecto, pero por falta de tiempo y recursos no se ha podido llevar a cabo ni una primera iteración.
Sistema de toma de decisión con Arboles de Decisión	Tras una evaluación más exhaustiva del proyecto no mostro que este sistema de decisión no era necesario para el mismo, por lo que procedimos a descartarlo.
Otras Consideraciones	
Acciones de la IA	Ciertas acciones de la IA como el sistema de combate a distancia no se han podido implementar por errores memoria que a falta de tiempo no se han podido solucionar, no hay zona afectada aparenta obviando la programación del combate.

Videojuegos II

Entregado	Explicación
Diseño e implementación de la arquitectura basada en componentes (diagrama de clases e implementación)	Arquitectura basada en clases diferentes las cuales llevan a cabo una función muy precisa y reducida. Lo que da la posibilidad que puedan realizar muchos modelos de los mismos objetos solo cambiando los componentes

Control del jugador por motor de físicas 2D, dynamic o kinematic	En nuestro juego se podía gestionar con un sistema 2d e implementamos procedimientos con el mismo.
Uso de trazados de rayos y otros tests de las físicas	Pruebas para conocer de forma más avanza el motor de físicas que damos uso
Mecánicas básicas entidades sin IA	Objetos como llaves para abrir puertas o monedas en forma de coleccionables
Mecánicas de acción	Nuestro juego tiene un sistema de combate sencillo, genero shoot and run.
Mecánicas de puzzle	Debes conseguir diferentes objetos para poder terminar los diferentes niveles
Formato propio para la definición de los niveles (documento con la especificación del formato)	Obligatorio, escoger un sistema y/o tipo de fichero para la creación y gestión de los niveles
Creación de un cargador de niveles	Obligatorio, necesario para la carga de los niveles
Implementación de cámara de seguimiento	En nuestro juego la cámara se halla fija al jugador.
Gestión de colisiones de la cámara de seguimiento	Las colisiones necesarias en nuestro sistema son con los límites del mapa para que no se muestren el espacio vacío.
Implementación de cámara inteligente que ajuste ángulo y zoom.	Se creó como entregable pero no necesario para el proyecto final.
Power-ups y elementos adicionales	Se realizó la implementación de estos poderes, pero se ha descartado para el producto final, se pueden ver implementados en hitos anteriores.
Integración de librería para GUI	Se ha integrado y utilizado para crear todas las interfaces del juego.
Diseño y creación de niveles	Se debían crear y diseñar niveles para tener un entorno donde jugar.
Implementación de menús	Se crearon los menús para poder moverse por las diferentes instancias o acceder al juego.
Implementación de HUD	Muestra las variaciones que ocurren o muestran valores de forma visual en el juego como la vida del jugador.
Parcial	
Sistema de depuración visual de las físicas	Se creó un sistema visual pero no cumple totalmente con lo requerido pues no representa todas las acciones físicas en pantalla
No entregado	
Gestión de colisiones de la cámara (sistema de depuración para ver colisiones)	Se creó una gestión de las colisiones, pero nos pasó por alto, el sistema de depuración realizando directamente la gestión final

Implementación de clipping (comparativa de FPS con/sin oclusiones)	Esta implementado el clipping en el motor, pero no se realizó la comparativa.
Sistema de Level-Of-Detail (LoD) para la selección de la malla grafica de los objetos	No se ha realizado ya que no tenemos visión de profundidad siendo innecesario la gestión.
Implementación de césped y vegetación	No se ha realizado por falta de tiempo y recursos
Implementación de sistemas de partículas	No se ha realizado por falta de tiempo y recursos

Postproducción Digital

Entregado	Explicación
Video con animación del logo de la empresa	Animación de nuestro logotipo
Cartel juego/proyecto	Muestra visual de nuestro juego, una primera vista de lo que se puede esperar de él.
HUD / Imágenes menús	Prototipos del HUD realizados con Photoshop
Tráiler juego/proyecto	Muestra el tráiler del juego con vídeos de otros juegos por falta de recursos de nuestro juego a la hora de realizar este entregable
Video final juego / explicación parte del proyecto	Breve vídeo donde se muestra una animación para cuando se ha finalizado el juego
Making of juego/proyecto	Vídeo de todas las etapas del curso y abp
Créditos	Vídeo con nuestros nombres y funciones en el desarrollo del juego

Proyectos Multimedia

Entregado	Explicación
Rellenar documento Gestión de riesgos.	Parte del proyecto obligatoria que nos enseña todo el control que se debe de tener de los proyectos del mundo de los videojuegos y el
Registrar de tiempos y % de realización de tareas en Project	
Revisar especificación proyecto	

Confeccionar informes de iteración e informe resumen de hito 1.	rigor que exigen los mismos.
Confeccionar informes de iteración e informe resumen de hito 2.	
Confeccionar informes de iteración e informe resumen de hito 3.	
Confeccionar informes de iteración e informe resumen de hito 4.	
Comparar la planificación prevista y real en Project hito 1	
Comparar la planificación prevista y real en Project hito 2	
Comparar la planificación prevista y real en Project hito 3	
Comparar la planificación prevista y real en Project hito 4	
Elaborar la presentación del Hito 1	
Exponer la presentación del Hito 1	
Elaborar la presentación del Hito 3	
Exponer la presentación del Hito 3	
Elaborar la presentación del Hito 4	
Exponer la presentación del Hito 4	
Detallar plan iteraciones del mes de noviembre. Creación subtareas y precedencias en Project	
Detallar plan iteraciones del mes de noviembre. Asignar recursos a las tareas en Project.	
Detallar plan iteraciones del mes de diciembre. Creación	

subtareas y precedencias en Project	
Detallar plan iteraciones del mes de diciembre. Asignar recursos a las tareas en Project.	
Detallar plan iteraciones del mes de enero. Creación Subtareas y precedencias en Project	
Detallar plan iteraciones del mes de enero. Asignar recursos a las tareas en Project.	
Detallar plan iteraciones del mes de febrero-marzo. Creación Subtareas y precedencias en Project	
Detallar plan iteraciones del mes de febrero-marzo. Asignar recursos a las tareas en Project.	
Detallar plan iteraciones del mes de abril-mayo. Creación Subtareas y precedencias en Project	
Detallar plan iteraciones del mes de abril-mayo. Asignar recursos a las tareas en Project.	
No entregado	
Aplicar el modelo EVA en Project	No evaluado

Realidad Virtual

Entregado	Explicación
Boceto de los personajes	Todos estos entregables son necesarios para crear un videojuego visualmente competente y poder realizar las animaciones y las interacciones del juego más realistas y consiguiendo una mejor inmersión en el mismo.
Bocetos de los elementos	
Bocetos del entorno	
Modelado de los personajes	

Texturizado de los personajes	
Rigging de los personajes	
Captura de movimiento	
Animación de los personajes	
Modelado del entorno y elementos	
Texturizado del entorno y elementos	

Técnicas avanzadas de gráficos

Entregado	Explicación
Visualización: visualizador OpenGL 4.X simple, con datos por programa. Shader básico.	Obligatorio y necesario para la implementación del motor.
Visualización: visualizador OpenGL 4.X simple, con datos por fichero. Shader básico.	Obligatorio y necesario para la implementación del motor.
Visualización: visualizador OpenGL 4.X de las entidades tipo malla. Shader básico. Integración con el motor. Sin materiales, texturas, cámaras ni luces.	Obligatorio y necesario para la implementación del motor.
Visualización: visualizador OpenGL 4.X de las entidades tipo malla. Con cámaras y luces de varios tipos. Visualización con registro de cámaras y luces.	Obligatorio y necesario para la implementación del motor.
Gestor de recursos: Parser de objetos en múltiples formatos, salida en modo texto.	Permite la lectura de diferentes formatos y así poder implementarlo con diferentes herramientas.
Gestor de recursos: Carga de modelos en formato múltiple en las entidades de tipo malla. Carga de varios ficheros para animación. Salida en modo texto.	Carga de varias mallas para las gestiones a posteriori de las animaciones por frames.
Árbol de la escena: Tipos de datos para nodos (completo)	Obligatorio y necesario para la implementación del motor.

y entidades (sólo clase padre; clases hijas sin contenido), construcción del árbol, recorrido del árbol, salida en modo texto.	
Árbol de la escena: Tipos de datos para entidades de tipo transformación (traslación, rotación y escalado relativos y absolutos. Otras transformaciones). Salida en modo texto.	Obligatorio y necesario para la implementación del motor.
Árbol de la escena: Tipos de datos para entidades de tipo cámara y luz (cámaras y luces de dos tipos). Salida de texto	Obligatorio y necesario para la implementación del motor.
Árbol de la escena: Tipos de datos para entidades de tipo de malla. Salida en modo texto.	Obligatorio y necesario para la implementación del motor.
Árbol de la escena: Tipos de datos para entidades de tipo animación (cuadro a cuadro). Salida en modo texto.	Programación para gestionar las mallas para realizar diferentes mallas
Visualización: visualizador OpenGL 4.X de las entidades tipo malla. Shader básico. Con cámaras y luces de varios tipos. Visualización con registro de cámaras y luces.	Obligatorio y necesario para la implementación del motor.
Gestor de recursos: Carga de materiales y texturas. Salida en modo texto	Obligatorio y necesario para la implementación del motor.
Visualización: visualizador OpenGL 4.X de las entidades tipo malla. Shader básico. Añadiendo materiales y texturas.	Obligatorio y necesario para la implementación del motor.
Fachada: creación de una fachada entre el motor y la aplicación.	Permite implementar el motor con diferentes proyectos convirtiéndola en una aplicación independiente.
Visualización: visualizador OpenGL 4.X con cuatro shaders avanzados	Shaders que nos permiten una mejor visualización de los objetos de escena y dando una mejora visual del proyecto
IGC: Aplicación: cargador de modelos, movimiento del modelo, de las luces y de la	Obligatorio y necesario para la implementación del motor. Integración del juego con nuestro motor.

cámara. Cambio de modelo, materiales y texturas. Animaciones. Integración con el motor; ICED: Integración con el videojuego, sustitución de Irrlicht por el motor propio	
Optimizaciones del motor (4 técnicas)	Optimiza el motor para una mejor gestión de los cálculos y recursos por parte de la tarjeta gráfica
Efectos visuales (3 efectos)	Nos permite crear efectos visuales específicos para nuestro proyecto
Extras (a determinar por el grupo)	Conjunto de extras para investigar por nosotros mismos y conseguir mejores resultados respecto al motor.

Negocios y Multimedia

Entregado	Explicación
Diseño funcional y visualización del proyecto: documentación con la especificación de todas las funcionalidades del sistema	Nos permite tener una visión de nuestro producto en el tratamiento de datos abiertos para realizar estudios y nuestro producto en el mercado. Así como ayudarnos a funcionalidades más atractivas para el público u otros de acogida.
Especificación y visualización de elementos y tecnologías a utilizar, por ejemplo, mediante la definición de una infografía del proyecto	
Contextualización del modelo de datos en las especificaciones del proyecto	
Definición de métricas e indicadores del proyecto. En el contexto indicado se definen los indicadores principales del proyecto	
Incorporación de open data, API's, y otras fuentes heterogéneas	
Modelo para la integración de datos (basado en lo visto en el punto anterior de fuentes de datos)	
Despliegue de la infraestructura del proyecto. Definición de cuadros de mando, implementación. Definición de KPI's implementación	
Valoración de los cuadros de mando y KPI's definidos e incorporación de nuevos elementos según datos	
Validación del funcionamiento	

Informes presenciales de seguimiento: presencial en sesiones de laboratorio todo el grupo	
---	--