

UADE

Entrega de primer trabajo práctico

Materia: Desarrollo Avanzado de videojuegos

Curso: 2do Año — Turno Noche

Modalidad de Entrega: Trabajo Práctico.

Cantidad máxima por grupo: 4 Participantes.

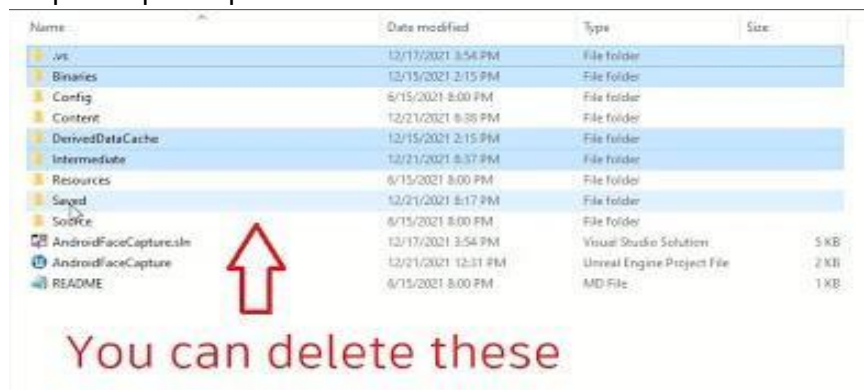
Esta entrega contempla el trabajo grupal de un proyecto único por equipo, realizado en el motor gráfico y de videojuegos Unreal Engine e implementando lógica a través del sistema blueprints.

Forma de presentación

Completamente obligatorio

- Se entregará una build y su proyecto correspondiente en un archivo .zip
- El proyecto contendrá solo las carpetas necesarias y no las creadas automáticamente por el motor.

Carpetas que se pueden borrar:



Name	Date modified	Type	Size
.vs	12/17/2021 3:54 PM	File folder	
Binaries	12/15/2021 2:15 PM	File folder	
Config	8/15/2021 8:00 PM	File folder	
Content	12/21/2021 8:38 PM	File folder	
DerivedDataCache	12/15/2021 2:15 PM	File folder	
Intermediate	12/21/2021 8:37 PM	File folder	
Resources	8/15/2021 8:00 PM	File folder	
Saved	12/21/2021 8:17 PM	File folder	
Source	8/15/2021 8:00 PM	File folder	
AndroidFaceCapture.sln	12/17/2021 3:54 PM	Visual Studio Solution	5 KB
AndroidFaceCapture	12/21/2021 12:31 PM	Unreal Engine Project File	2 KB
README	8/15/2021 8:00 PM	MD File	1 KB

You can delete these

- El proyecto debe haber sido realizado en la versión del motor que está instalada en las computadoras de la institución.
- La build debe funcionar completa y correctamente, sin ninguna excepción de referencia nula o excepción que impida el funcionamiento correcto del juego.

De no cumplirse este último punto, la entrega queda **desaprobada automáticamente**, sin ningún tipo de excepción.

- La build y proyecto que se toma en cuenta es la que está cargada en el horario que inicia la entrega.
- Es obligatorio un conocimiento prácticamente completo del funcionamiento de la entrega por parte de todos los participantes de un grupo. De comprenderlo necesario, el profesor puede hacer preguntas a nivel individual para verificar el conocimiento de uno o varios participantes en particular.
- La nota **no es grupal** y pueden haber participantes reprobados si no demuestran un aporte significativo a la entrega (el nivel exigido para el aporte y la decisión de si dicho aporte está correctamente implementado se rige por criterio del profesor).
- Se contempla como lógica solo lo implementado por los participantes del grupo. Todo contenido y lógica implementada por terceros (ya sea descargados de internet, conocidos o assets del marketplace), será tomado en cuenta como un extra, pero no para la consigna principal y la nota mínima de aprobación.

Consigna Principal

Completamente Obligatorio - 4 Puntos (si está perfecto)

- Se debe crear un juego, utilizando el motor Unreal Engine y contemplando un movimiento en 3D (está permitido utilizar los proyectos plantilla de 3ra persona, 1ra persona y Top-Down como base).
- El juego debe contener un menú principal, un nivel de longitud media y complejidad similar a los mostrados en clase, pantalla de victoria y de derrota.
- El proyecto debe cumplir con la definición básica de juego, la misma siendo compuesta de los puntos:
 - Hay un objetivo y es comunicado al jugador.
 - Condición de victoria y derrota (el juego no puede ser un endless runner).
 - Múltiples interacciones con el mundo que presentan un desafío al jugador.
 - Hay una zona de “cero peligro” en donde el jugador no pierde, esto puede ser un tutorial o una zona previa al nivel en donde los enemigos no atacan al jugador y el mismo puede aprender las mecánicas a salvo.
 - De ser un nivel a parte, el tutorial se presenta antes del nivel principal.
- Todas las acciones/verbos del jugador deben tener animaciones correctamente aplicadas.
- Todas las acciones/verbos del jugador deben tener sonidos correctamente aplicados.
- Debe haber un enemigo, el mismo debe ser un Pawn o peón que implemente percepción.
- Se debe instanciar, como mínimo, un actor en el nivel (no cuenta lo instanciado por lógica ya hecha en la plantilla).
- Deben haber, como mínimo, 2 actores que contemplen una interacción con el personaje. Dentro de los mismos no se cuenta el enemigo.

Ejemplos de estos actores (la lista es ejemplificante, no tiene función limitante):

- Trampolín
- Puerta
- Ascensor
- Palanca o *Lever* (que tenga un efecto en el mundo frente a su interacción).

(Una puerta abierta por palanca o cualquier combinación de este estilo se toma como un solo actor de la cantidad pedida).

Se puede reemplazar uno de estos actores interactivables con una mecánica extra (que implemente lógica de blueprint). Ejemplos:

- Desplazamiento o *Dash*
- Gancho o *Grappling hook*
- Subida de repisa/saliente o *Ledge Climb*
- El nivel debe tener un diseño coherente, presentando un whitebox (prototipado con herramientas de modelado y sin necesidad de todos los assets y materiales aplicados).
- Debe incluirse un material personalizado que implemente más que solo mapas de texturas.
- Se debe implementar una serie de cheats.
 - Tecla F1 -> GodMode (el jugador es inmortal, no puede morir).
 - Estilo toggle. Debe encenderse y apagarse cuando se presiona el input.
 - Tecla F2 -> Flash (La velocidad del jugador incrementa considerablemente).
 - Estilo toggle. Debe encenderse y apagarse cuando se presiona el input.

Una vez **completa la consigna obligatoria y cumplida la condición mínima de aprobación**, se pueden sumar puntos a través de los siguientes adicionales (la nota máxima es un 10, por más que los puntos extra sumen más):

2 Puntos (si está perfecto)

- El tutorial se carga la primera vez que se reproduce el juego. Pero, una vez superado el mismo, la siguiente vez que el jugador presiona el botón "jugar", se presenta una pantalla que permite decidir al jugador si desea repetir el tutorial o saltarlo y pasar directamente al juego principal.

2 Puntos (si está perfecto)

- 1 Enemigo extra. Con lógica y modelo distintos.

2 Puntos (si está perfecto)

- 1 nivel extra, con un diseño o *layout* diferente.

2 Puntos (si está perfecto)

- 2 materiales personalizados extras, con efectos especiales o efectos de partículas (de alto nivel de calidad). Ejemplos:
 - Agua o *Water shader*
 - Portal
 - Hojas cayendo
 - Polvo al caminar
 - Líneas de velocidad (estilo anime)

1 Punto (si está perfecto)

- 1 actor interactuable extra. Con lógica y modelo distintos.

1 Punto (si está perfecto)

- Todo texto se encuentra en inglés.

Aclaraciones

- El/los enemigos deben presentar un desafío real y no ser pasivos ante el jugador (No se acepta que los enemigos simplemente se muevan y mueran al ser atacados)
- El loop jugable debe tener sentido y exigir al jugador la utilización de todas las mecánicas implementadas (las mecánicas fantasma o "la mecánica a mí me gusta y el jugador le debería dar uso" resta puntos).
- Debe haber una progresión comprensible y un gameplay loop que sea digno de un juego real.
- Los errores de ortografía bajan la nota (a criterio del profesor).
- Se debe demostrar un correcto conocimiento e implementación de todos los conceptos vistos en la cursada. De encontrarlo pertinente, el profesor puede pedir explicación o respuesta de un tema que no encuentre bien implementado o comprendido en el proyecto y que esta explicación afecte a la nota y condición de aprobado o desaprobado de la entrega.