



Parcial 2

# Diseño y Programación de Videojuegos

Materia: Programación 2

**Profesor Contenidista:** Lorenzo Caballero Sollmann

# Programación 2 - TP 2

Tercer cuatrimestre

Siguiendo el juego realizado en el TP1, el objetivo será agregarle algunas cosas y pulir el contenido con criterio para mantener el peso y el rendimiento del juego en valores aceptables.

## Consignas base a cumplir (del TP 1):

- Debe ser si o si en Unity 3D
- Menú principal con botones para cerrar el juego, empezar a jugar y mostrar controles
- Mostrar en alguna parte del menú principal el nombre de cada integrante del equipo
- Deberá haber una cantidad de enemigos y/u obstáculos complejos igual a la cantidad de integrantes del grupo más uno
- Debe usar Animator Controller con varios estados y transiciones en por lo menos un objeto del juego
- Una escena de prueba para el prototipo, creada usando ProBuilder y figuras primitivas de Unity, aunque NO pueden ser solo figuras primitivas
- Música y efectos sonoros placeholder para dar feedback en las acciones más importantes

## Consignas propias del TP 2:

- Deberá tener resueltas las correcciones del TP1
- Se deberá armar un nivel
  - Deberá poder accederse al nivel de prueba/prototipado
  - El nivel no debe superar los 5 minutos de duración aproximadamente
- Deberán poner efectos de sonido por lo menos en las acciones e interacciones principales como así también música. Todo el audio implementado en el juego deberá ser controlado por un Audio Mixer
- Deberán tener por lo menos tres sistemas de partículas diferentes en alguna parte del proyecto (ej: efectos de daño, impactos de proyectiles contra las paredes, explosiones, efectos como el de un lanzallamas, curaciones, etc.)

- Armar partículas propias con algo de trabajo, aunque no se vean tan bien, va a ser más apreciado en la nota que usar partículas descargadas de Internet
- Deberá usarse iluminación de algún tipo en el proyecto, pero teniendo muy en cuenta conceptos de optimización (ej: no usar demasiadas luces y sombras, apagar por código las luces lejanas o usar luces bakeadas)
  - Usar únicamente color de luz del entorno sin ningún tipo de luces en escena también es una opción aceptable, dependiendo de la estética del juego

## Restricciones:

- No se podrán usar modelos placeholder (ej: formas por defecto de Unity o ProBuilder) para representar ningún elemento del juego (en el caso de que realmente no se justifique usar un modelo puede consultar con el profesor). En vez de eso, se deberán hacer o buscar en Internet modelos más o menos adecuados al estilo visual del juego
  - Si descarga paquetes del Asset Store, debe eliminar todo lo que no use para no sobrecargar el tamaño del juego al momento de la entrega
- Se deberá cuidar de achicar las texturas usadas en los objetos para que ocupen lo menos posible sin verse mal, tanto en build (cambiando los import settings de las texturas) como en el proyecto y Library (escalando las texturas en GIMP, Photoshop o similares al tamaño decidido en los import settings de la textura)
  - Como regla general, y estimando el largo y la complejidad de un proyecto promedio, no deberían hacer falta varios gigas de texturas salvo casos particulares
- Si el proyecto lo amerita, se podrán usar elementos como skyboxes, reflection probes y niebla, pero siempre teniendo en cuenta los costos de proceso para que el framerate no baje demasiado
- Se aceptará el uso de post-procesos de cámara en caso de conocerlos y saber usarlos, pero siempre y cuando los efectos no comprometan drásticamente al rendimiento del juego

## Criterios de desaprobación:

Cualquier error que se genere en el código e impida que el juego se ejecute correctamente **se deberá corregir para aprobar.**

Cualquier proyecto que tenga bajones serios de framerate en cualquier parte del juego, ya sea por animators, scripts, cantidad de objetos a

dibujar, luces, post-procesos o cualquier otro elemento, **se deberá corregir y optimizar para aprobar.**

Si el flujo de juego se rompiera en algún momento (ej: que el personaje traspase el terreno del nivel y caiga eternamente al vacío, o que un enemigo que tenemos que derrotar spawnee afuera del nivel y sea inalcanzable, impidiéndonos ganar), y aunque no sea por un error de código sino por uno de lógica, **los errores deberán corregirse para aprobar**

Al igual que en el primer parcial, el uso de ejemplos de clase en la totalidad del proyecto, o gran parte, puede restar puntos.

**Aclaración importante:** Si bien el trabajo es grupal, cada alumno según el criterio del docente podrá tener notas distintas. Es imperioso que independientemente de quien haya desarrollado alguna parte del código, todos los integrantes del grupo sean capaces de defenderlo y explicarlo.

En caso de que el docente lo considere pertinente, podrá desaprobar a algún integrante sin necesidad de desaprobar al resto del grupo.

### **Formato de entrega:**

Versión de Unity: 2022.3.5f1

Deberá entregarse el proyecto de Unity comprimido en una archivo de extensión .rar o .zip via google classroom. El archivo deberá contener en su interior todas las carpetas del proyecto, y en el nombre del archivo deberá figurar el TP y el apellido de cada integrante del grupo (ej: "TP1 - Apellido1,Apellido2.rar").

En caso de que pese más de 1GB, se sugiere no agregar al archivo comprimido la carpeta Library, sino subirla por separado en otro archivo comprimido.

## Formato de entrega:

- Se entregará un archivo comprimido (ZIP, RAR, 7Z, etc) con los siguientes archivos:
  - El proyecto entero de Unity (incluyendo Library)
    - En caso de que el proyecto pese más de 1GB, se recomienda comprimir Library en un archivo aparte y subirlo por separado
  - Una build ejecutable del juego en formato EXE
  - Un TXT con el nombre completo de cada integrante del equipo
- **NOTA:** Para reiterar el primer punto, no se aceptarán proyectos por medio de archivos .unitypackage, ni a los que les falten las carpetas imprescindibles para poder abrirlos en forma correcta (Assets, ProjectSettings, Packages y Library)
- El archivo comprimido deberá contener el nombre del TP y el apellido de cada integrante del equipo (ej: TP1 - Apellido1,Apellido2.zip)
- El proyecto deberá estar hecho en Unity 2022.3.5f1. **No se aceptarán proyectos en otras versiones**
- Los proyectos se deberán entregar por medio del Classroom, en la tarea del TP2
  - Alcanza con que una persona por equipo suba el proyecto
  - En caso de tener problemas para subir el proyecto directamente al Classroom, se podrá subir a un drive o hosting propio, y subir al Classroom un txt con los vínculos a los archivos
  - En caso de subir versiones con correcciones, recuerden enviar un mensaje al classroom indicando que lo hicieron, para facilitar el rastreo y la corrección de los proyectos que tengan correcciones hechas