

# Programación 1 - TP Final

Segundo cuatrimestre 2023

## Consignas:

Se debe continuar el desarrollo del videojuego realizado como cierre de proyecto de segundo cuatrimestre.

Dicho video juego deberá tener completa la consigna del segundo parcial (por consiguiente, la del primero también), tener los cambios marcados por el docente y agregar algunos elementos extra (a definir a continuación).

En el caso de que deseen cambiar de juego, deberán cumplir con las consignas del TP1 y TP2.

Es importante que se respeten todas las consignas de entrega.

El videojuego contará con los siguientes requisitos:

- Tener resueltos los errores mencionados durante el TP2
- Agregar pantalla de créditos o incluirlos al menú
- Agregar música en los menús
- Agregar opción de rejugaridad en la pantalla de victoria o derrota
- Nuevos features que sumen al menos **10 puntos** de la lista de abajo
  - Serán ampliamente valorados los trabajos que incluyan elementos desarrollados en la materia Tratamiento Digital de la Imagen 2
  - Aunque se valora que haya variedad en los ítems elegidos, se puede repetir un mismo ítem varias veces (ej: 2 enemigos y 1 power-up), siempre y cuando sean suficientemente distintos entre sí
  - En el caso de que el trabajo de cursada no se acomode a los ítems pedidos en el TP, se recomienda hablar con el docente a cargo de la mesa o con quien se haya cursado para definir ítems alternativos. Deberá entregar una copia de los mails donde se definieron
  - Si algún ítem se aplica como mejora sobre algún objeto ya existente, valdrá la mitad de puntaje
    - Ej: agregar un nuevo comportamiento no-trivial a un enemigo preexistente sumaría dos puntos en vez de cuatro
- Entregar un PDF explicando brevemente la lista de ítems elegidos y cómo funcionan
  - **Ejemplo:** “Nuevo enemigo (4p): Se agregó un fantasma invencible, que aparece después de cierto tiempo y persigue lentamente al jugador, atravesando paredes y derrotándolo instantáneamente si lo toca”

## Listado de features posibles:

Dada la versatilidad de los diferentes juegos desarrollados en el segundo trabajo práctico, se da la posibilidad de elegir sobre una lista de elementos los que deseen desarrollar para su final (**[IMPORTANTE]** más allá de los desarrollados para el segundo parcial). Cada elemento cuenta con un puntaje aproximado según su dificultad para desarrollarlo, el total de la suma de los elementos seleccionados debe alcanzar por lo menos los 10 puntos.

- **Nuevo Enemigo (5p)** : implica desarrollar un nuevo enemigo con nuevas mecánicas + animación + feedback audiovisual.
- **Nueva mecánica para el personaje principal (4p)**: Implica desarrollar una nueva mecánica para el personaje principal, como por ejemplo un nuevo tipo de disparo con su nuevo comportamiento, o una nueva skill, etc. Dicha mecánica deberá contar con su feedback audiovisual.
- **Power Up (2p)**: Implica desarrollar un nuevo power up para el personaje principal, por ejemplo boost de velocidad, boost de potencia de disparos, etc. Dicho power up deberá dar feedback audiovisual.
- **Nuevos obstáculos con nuevos comportamientos (2p)**: Implica desarrollar un nuevo obstáculo con algún nuevo comportamiento en el nivel. Dicho obstáculo deberá dar feedback al usuario cuando se interacciona con él.
- **Nueva mecánica de gameplay (3p)**: Implica desarrollar una nueva mecánica para el gameplay, como por ejemplo tiempo, portales, checkpoints, etc. Dicha mecánica deberá dar feedback audiovisual cuando se interacciona con ella.

**[IMPORTANTE]** En el caso de que no llegue a los diez puntos requeridos pero alcance por lo menos ocho puntos, puede ser considerado como válido si los elementos, mecánicas u obstáculos integrados superan las expectativas.

## 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 en formato Windows (.exe), con todos los archivos necesarios para poder usarla (OBLIGATORIO)
  - Un TXT con el nombre completo de cada integrante del equipo
  - Archivos PDF con:
    - Estas consignas
    - La devolución del docente recibida en el TP 2
    - La explicación y detalle de los ítems elegidos para el final
- **NOTA:** Para reiterar el primer punto, no se aceptarán proyectos por medio de archivos .unitypackage, ni en los que les falten las carpetas imprescindibles para poder abrirlos en forma correcta (Assets, ProjectSettings, Packages y Library)
- El nombre del archivo debe ser TP Final - Apellidos y nombres
- El proyecto deberá estar hecho en Unity 2021.3.12. **No se aceptarán proyectos en otras versiones**
- Los proyectos se deberán entregar por medio del DV Panel
  - Alcanza con que una persona por equipo suba el proyecto

## Consignas base a cumplir del TP 1:

El juego presentado deberá tener las siguientes características:

- Estar desarrollado en Unity 2D (sin excepciones)
- El juego debe ser top-down (pueden tomar como referencia Binding of Isaac, Micro Machines, Zelda, naves, Asteroids, Arkanoid, etc.)
- Deberá contener una escena de prueba donde se demuestren las mecánicas bases del juego:
  - Movimiento del personaje
  - Interacción con el entorno y entidades (por ejemplo: proyectiles, agarrar ítems, dialogar, etc.)
  - Sistemas de energía (HP) para personaje y enemigos (si hubiera combate)
- Deberán crearse al menos dos enemigos u obstáculos (se recomienda para profundizar sobre el aprendizaje un enemigo y un obstáculo), ambos con comportamientos relativamente simples pero no triviales, como ser:
  - Perseguir al jugador
  - Moverse entre puntos previamente determinados
  - Disparar hacia el personaje
  - Cambiar de dirección al chocar contra paredes
  - NOTA: No cuentan para esta consigna objetos que tengan comportamientos excesivamente simples (ej: pinches estáticos)
- Entre el jugador y los enemigos, al menos uno de ellos (o ambos) debe poder instanciar algún tipo de objeto:
  - Disparos
  - Drops (pociones curativas, llaves, etc.)
  - Otros enemigos (por ejemplo un slime que al morir se divide en dos) ●

Usar Debug.Log para demostrar el resultado de todas las acciones importantes: ○

Ganar o perder puntos

- Recibir daño o curarnos
- Muerte de enemigos
- Diálogos
- Conseguir llaves, pociones, etc.
- Si se usan gráficos para objetos que deban estar adelante o atrás de otros, deberán asignarse los Sorting Layers o valores de Order in Layer para que siempre se vean correctamente
- El nivel puede armarse con Tiles, usando la Tile Palette in
  - Se recomienda que los objetos dinámicos (por ejemplo puertas) no sean tiles, sino prefabs ubicados a mano
- Los valores numéricos del juego se deberán poder manejar por variables públicas (por ejemplo: HP, velocidad de movimiento, daño de ataque, etc.)

## Consignas base a cumplir del TP 2:

- **Tener resueltos los errores mencionados durante el TP1**
- **Niveles:** 1 nivel completo, con condiciones de victoria y derrota (no se evaluará con profundidad el diseño del nivel). Aparte, debe estar disponible el nivel de prueba/prototipado del TP1 para acceder desde el menú principal.
- **Pantallas y menús:**

- Pantalla principal, con opciones para:
  - Arrancar el juego (tanto el nivel prototipo como el nuevo)
  - Ver instrucciones de juego y controles (se recomienda mostrarlas antes de arrancar a jugar)
  - Salir del juego
- Nivel de juego + nivel de prototipado
- Pantalla de victoria y derrota con opción para volver al menú
- **Arte:** Se valorará que el juego integre el arte desarrollado durante el cuatrimestre, salvo algunos elementos como efectos especiales, explosiones, partículas, etc, que no serán obligatorias.
  - El juego deberá contener animaciones y uso de Animator con varios estados, parámetros y transiciones en al menos una entidad (PJ, enemigo, etc.)
  - El juego deberá contar con efectos de sonido para las acciones principales (ej: recibir daño, disparar, agarrar items, etc.) y música de fondo en el nivel
- **Gameplay:**
  - Imprimir mensajes en consola no será aceptado como mecanismo de feedback