

# Python

## CONSIDERACIONES

Leer el orientador de la clase y haber completado todas las Guía

## Trabajo Integrador - Videojuegos

Objetivo: Aplicar todo lo aprendido hasta el momento para resolver las diferentes problemáticas que presenta en Trabajo Integrador. Al terminar podrá implementar un sistema de colisiones

Temas: Entradas y Salidas, Librerías pygame, instrucciones al uso de clases y objetos

### A resolver

#### La entrega consta de 2 partes:

- 1- Escrito académico: Se plantean los elementos del juego junto a una breve descripción de las mecánicas. Se complementa con un diseño estático de todo el nivel
- 2- Prototipo: Desarrollo de un nivel del juego, la entrega debe ser funcional y libre de errores.

### Descripción:

En este trabajo, tendrás la oportunidad de aplicar los conceptos aprendidos sobre desarrollo de videojuegos para crear tu propio nivel básico. El objetivo es que pongas en práctica tu creatividad y conocimientos, desarrollando un nivel jugable y entretenido. Deberás presentar un escrito académico con el siguiente contenido:

1. Deberás presentar un escrito académico con el diseño del nivel, incluye capturas de pantalla, diagramas o ilustraciones que muestren el diseño y la disposición de los elementos en tu nivel. Acompaña tu presentación con una descripción detallada de las decisiones tomadas y los objetivos del nivel.
2. Realiza una reflexión final sobre tu diseño de nivel. Evalúa su jugabilidad, los puntos fuertes y débiles, y las posibles mejoras que podrías implementar en futuras versiones.

Recuerda que este trabajo tiene un enfoque libre, por lo que puedes dejar volar tu imaginación y experimentar con diferentes ideas. No olvides utilizar los conceptos y elementos del diseño de niveles que hemos estudiado durante el curso. ¡Diviértete creando tu propio nivel y disfruta del proceso de diseño!

Como disparador y como insumo para armar el escrito académico, puedes tener en cuenta los siguientes puntos

1. Tema y género:

- Elige un tema y un género para tu videojuego. Puede ser una aventura de plataformas, un juego endless, un rompecabezas, etc. Asegúrate de que el tema y el género sean coherentes y se ajusten a tus intereses.

2. Objetivos del nivel:

- Define los objetivos que los jugadores deben alcanzar en tu nivel. Puede ser llegar a un punto específico, vencer a un jefe, resolver un puzzle, recolectar objetos, etc. Estos objetivos deben ser alcanzables y desafiantes al mismo tiempo.

3. Diseño del nivel:

- Crea el diseño de tu nivel utilizando papel y lápiz, o cualquier software de diseño de niveles que prefieras. Considera aspectos como la disposición de plataformas, la ubicación de obstáculos y enemigos, la distribución de power-ups o tesoros, etc.

4. Elementos del nivel:

- Utiliza los elementos del diseño de niveles aprendidos, como terreno y ambiente, objetos y obstáculos, enemigos y AI, puzzles y secretos, para construir tu nivel. Asegúrate de que estos elementos se ajusten al tema y género de tu juego.

5. Progresión y dificultad:

- Diseña una progresión adecuada en la dificultad a lo largo del nivel. Asegúrate de que los desafíos aumenten gradualmente y que exista un equilibrio entre momentos de acción, exploración y resolución de puzzles.

Junto al escrito deberás entregar el código usando el siguiente formato:

## ORGANIZACIÓN DEL PROYECTO

- Arquitectura del Proyecto:
- Nombre\_del\_Proyecto
  - lib
    - \_\_init\_\_.py
    - Color.py
    - Core.py

- Var.py
- (otras librerías)
- Sprite
  - Carpetas de animaciones
  - Enemigos obstaculos
  - Personaje
  - Enemigos / obstáculos
  - Fondos de pantallas
- Main.py
- Creditos.py

#### descripción

- Lib: Módulo que continúe todas las librerías del juego
- Color.py: lista de constantes con los colores de la paleta elegida para el juego
- Core.py: conjunto de todas las clases que se desarrollaron en el juego
- Var.py: lista de variables globales
- (otras librerías): conjunto de archivos con las librerías propias que hayas desarrollado para tu juego.
- Sprite: carpeta con todos los Sprite del juego, para organizar el contenido creamos carpetas para cada personaje/elemento con sus animaciones
- Main.py: punto de acceso al juego.
- Créditos.py: archivo con comentarios `''' --- '''` con información del juego, nombre de los miembros del equipo, y datos de la cursada