

# REUNIÓN CON EQUIPO ARTISTICO

## 09-10-2025

| Integrantes: [ 3 ] | Asistencias: [ 3 ] | Confirmados: [ 3 ] |

La reunión ha sido para poner al día respecto a los cambios realizados sobre la temática del juego, tipos de mapa, objetos necesarios...

### Idea General:

La duración del reto serán **5 semanas**. En total **15 horas**.

<u>HITO 1: PLANIFICACIÓN</u> 13/10/2025	<u>HITO 2: CONSTRUCCIÓN</u> 10/11/2025	<u>HITO 3: PRESENTACIÓN</u> 14/11/2025
Documento de diseño preliminar	Código fuente	Documentos de presentación
	Documento de diseño final	

El escenario de los niveles empezará y terminará en un bosque. Entremedias habrá fondos de cueva mezclándose con elementos de una fábrica.

#### - Bosque:

- El primer fondo estará en buen estado. Verde, repleto de naturaleza con colores vivos.
- El segundo y último estará moribundo. Troncos secos, rotos, vegetación muerta. Con colores grises y oscuros o apagados.
- Elementos como árboles, arbustos, piedras con musgo, hierbajos, troncos...

#### - Cueva:

- El primer fondo será una cueva llena de vegetación. Con colores vivos.
- Los intermedios irán deteriorándose o ensuciándose, mostrando más elementos de residuos u objetos relacionados a fábricas.
- El ultimo estará repleto de elementos relacionado con la tecnología y estará más sucio.
- Elementos como piedras grandes y pequeñas, estalactitas, estalagmitas, piedras sucias, cajas, tuberías, cadenas, cuerdas, escaleras de mano...

---

### Fondos:

- **Bosque:**
    - Bosque vivo:

Lleno de **árboles**, puede que **lago** al fondo, **vegetación viva**. Que la iluminación sea de la **Golden hour** justo antes del atardecer. Para que el periodo dentro de la cueva transcurra durante el atardecer.
    - Bosque moribundo:

**Troncos cortados**, secos. Con **niebla toxica**, **residuos** y de fondo que aparezca alguna **fabrica** y/o una **salida de residuos**. Mostrando que es la causa del estado del bosque. La iluminación puede ser tanto de un **atardecer a punto de terminar o noche temprana**.
  - **Cueva:**
    - Entrada:

La entrada tiene que tener algo de **iluminación que venga desde el primer fondo**. Con elementos de vegetación como **enredaderas, hierbajos, musgo...**
    - Entremedias:

Los fondos tienen que **mostrar una transición**. Las primeras partes, tienen que estar en **mejor estado y según avance**, tiene que comenzar a tener más **elementos tóxicos o sucios**, relacionado con una fábrica. Niebla, rocas sucias por objetos pegajosos, cajas, barriles... La iluminación tiene que ser **ligeramente oscura**, mezclándose con entradas permitiendo **luz natural y artificial**. Pudiendo ser iluminada con faroles, lámparas de techo...
    - Salida:

En el fondo de la salida se tiene que ver que la **cueva está integrada, con mecanismos o elementos** parecidos, a una fábrica. Teniendo en cuenta que la parte de **la salida tiene que ser parte de una fábrica**. Como un pasillo de salida o entrada de mantenimiento. La iluminación tiene que ser mayoritariamente generada por elementos artificiales.
- 

## Elementos de Decoración:

**Elementos de decoración** que serán colocados **entre el fondo y el plano central** donde anda el personaje y **entre plano central y la cámara**.

- **Bosque:**
  - Arboles:

Especie a elección. Sanos, moribundos y muertos.

    - *Troncos.*
    - *Árbol completo.*
    - *Ramas.*
  - Arbustos:

Especie a elección. Sin y con flores. En buen estado, pisados, sucios, quemados...

    - *Individual.*

- *Grupos.*
  - **Cueva:**
    - Piedras:

Tras **entrar en contacto** con el jugador **desaparece** en los segundos marcados.

      - Piedras genéricas individuales.
      - Piedras agrupadas.
      - *Estalactitas y estalagmitas.*
    - Columnas:

Tanto de **madera** como **piedra** o **acero**. Tanto robustas como frágiles.

      - *Tablas de madera.*
      - *Troncos.*
      - *Superficie lisa de piedra.*
    - Enredaderas:

Enredaderas tanto **para el fondo** como para colocar **enfrente** de la cámara.
  - **Artificiales:**
    - Cajas y barriles:

**Localizaciones** donde el **jugador reaparecerá** de forma **automática** o en caso de que muera. El inicio de cada nivel.
    - Columnas:

Tanto de **madera** como **piedra** o **acero**. Tanto robustas como frágiles.

      - *Tablas de madera.*
      - *Superficie lisa de piedra.*
      - *Acero.*
    - Cables y cuerdas:

Cables y cuerdas tanto **para el fondo** como para colocar **enfrente** de la cámara.
    - Escaleras:

Escaleras de mano para colocar **verticalmente** mirando a la cámara.

      - *De madera.*
      - *De metal.*
      - *Mezcla.*
    - Tuberías:

Tuberías metálicas en varios estados para ponerlas tanto de fondo como en el frente.

      - *Enteras.*
      - *Con grietas.*
      - *Con fluidos cayendo (Este tendrá que tener mínimo 2 variantes para animar que esta cayendo).*
- 

**Objetos:**

**Elementos interactivables** que serán **colocados en el plano central**. Estos se colocarán tanto **horizontal** (cara mirando arriba) como **verticalmente** (cara mirando a la cámara).

- **Botones y palancas:**

- Botones:

- Pulsadores que tienen que tener **dos estados** diferentes.

- *Pulsado.*
    - *Sin pulsar.*

- Palancas:

- Palancas que tienen que tener **mínimo dos estados** diferentes.

- *Arriba / Atrás.*
    - *Centro.*
    - *Abajo / Enfrente.*

- **Cajas y barriles:**

- Solidas:

- Cajas de madera que se usarán para **empujar**.

- *Enteros*
    - *Deteriorados.*
    - *Sucios.*

- Barriles:

- Metálicos.

- *Enteros*
    - *Deteriorados.*
    - *Rotos.*

- **Puertas y paredes:**

- Puertas:

- Puertas de **madera o metal**. Mínimo necesitan dos estados.

- *Picaporte quieto (cerrada).*
    - *Picaporte movido (abriéndose).*
    - *Entera.*
    - *Rota.*

- Paredes:

- Paredes con **grietas**. Tienen que tener *variaciones* para hacer que se está **rompiendo**.

- *Entera (Con grietas).*
    - *Rompiéndose.*
    - *Rota.*

- **Ascensores:**

- Paredes:

- Tres paredes que se colocarán en los cuatro lados de lo que formaría el ascensor. Tienen que valer para que

- *Enteros*

- Puerta:  
Cajas de madera que se usarán para **empujar**.
    - *Enteros*
  - Suelo:  
Metálicos.
    - *Enteros*
- 

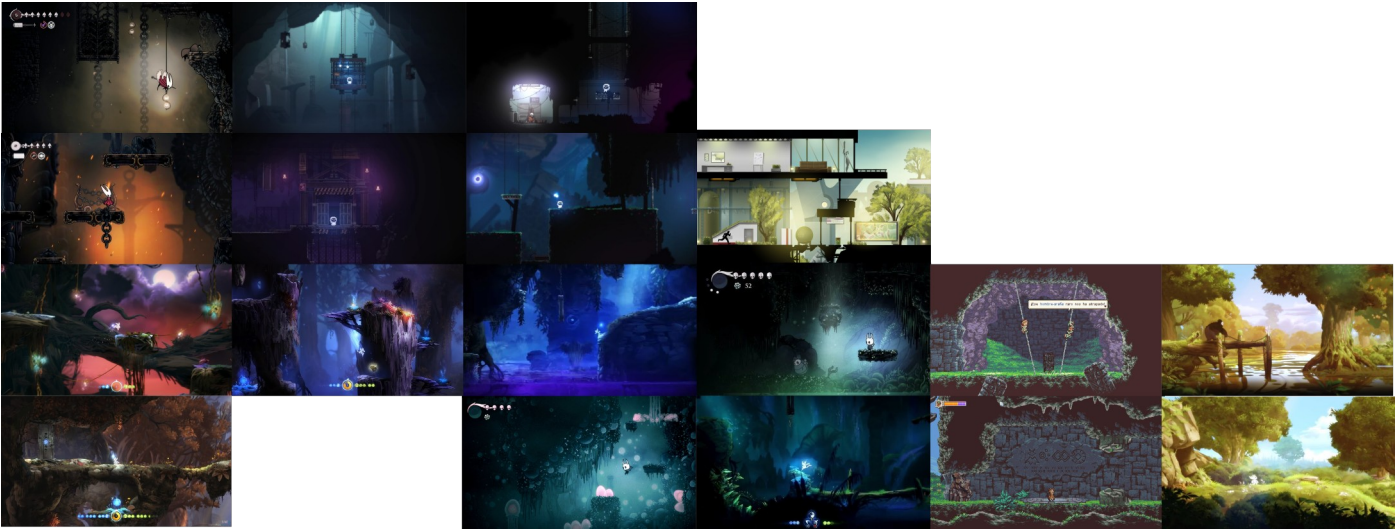
## Menús:

Los menús necesitan un **fondo y logos o símbolos**. Mínimo se necesitará un **fondo para el menú principal** y uno distinto para el de **final de juego** (Game Over) y se pueden reutilizar. También hay que **organizar y estructurar** donde estarán colocados los botones u opciones, título...

- **Menú Principal:**
    - Start:  
Comienza el juego.
  - **Menú de Pausa:**
    - Resume:  
Vuelve al juego.
  - **Menú de Game Over:**
    - Reintentar:  
**Reinicia** el juego empezando **desde la Escena Nº1**.
    - Volver al Menú:  
**Vuelve** a la pantalla del **menú principal**.
  - **Pantalla de Fin:**  
Pantalla tras **terminar el juego** que **vuelve al menú principal** automáticamente.
-

## Referencias:

- Hollow Knight.
- Silksong.
- Owlboy.
- Sheepy: A Short Adventure.
- Ori and the Blind Forest.
- Ori and the Will of the Wisps.
- Vector.



## Rubrica:

COMPETENCIAS TÉCNICAS PMD					
<b>Diseño Juego 20% RA5</b>	Escenario básico con pocos elementos. No hay una temática clara ni narrativa. Contiene menos de 5 objetos.	Escenario funcional con al menos 5 objetos interactivos o decorativos. Temática poco definida, narrativa básica.	Escenario completo con al menos 10 objetos bien distribuidos. Temática definida y narrativa simple que da contexto al juego.	Escenario detallado con al menos 20 objetos, incluyendo elementos interactivos y decorativos. Narrativa clara y coherente que guía el objetivo del juego (Puede dividirse en niveles)	Escenario altamente detallado con más de 30 objetos (interactivos y decorativos), bien organizados y diseñados. Temática original y narrativa inmersiva.
<b>Mecánicas y Jugabilidad 20%</b>	Implementa al menos 1 mecánica básica funcional (por ejemplo, movimiento simple). Colisiones mínimas configuradas.	Implementa al menos 2 mecánicas funcionales (por ejemplo, movimiento y salto). Colisiones básicas que funcionan correctamente.	Implementa al menos 3 mecánicas funcionales (por ejemplo, movimiento, salto y recoger objetos). Colisiones bien configuradas para objetos relevantes.	Implementa 4 o más mecánicas, incluyendo interacciones con objetos (activar, recoger, mover). Las colisiones y físicas están configuradas correctamente en todo el entorno.	Implementa 5 o más mecánicas, incluyendo cambios de comportamiento (velocidad, habilidades, modos de movimiento) y mecánicas originales o creativas.
<b>Animaciones 15%</b>	No se ha diseñado ningún NI-vel	Implementa al menos 1 animación funcional básica (por ejemplo, caminar o correr). No utiliza máquina de estados.	Implementa al menos 2 animaciones funcionales (por ejemplo, caminar y saltar), con transiciones básicas configuradas manualmente.	Implementa al menos 3 animaciones funcionales y utiliza una máquina de estados (Animator) básica para gestionar las transiciones.	Implementa 4 o más animaciones, con transiciones configuradas correctamente en el Animator (por ejemplo, caminar, saltar y atacar). Utiliza parámetros como triggers o bools para controlar las animaciones.
<b>Interfaz Usuario UI 15%</b>	Implementa un menú inicial básico con al menos 1 botón funcional (por ejemplo, iniciar juego).	Menú inicial con al menos 2 botones funcionales (iniciar juego y salir). Interfaz básica durante el juego (por ejemplo, contador de puntos fijo). sin animación	Menú inicial completo con al menos 3 botones funcionales (iniciar juego, créditos, salir). Sistema de puntuación o contador dinámico implementado correctamente.	Menú inicial con diseño atractivo y funcionalidad completa (iniciar, créditos, opciones, salir). Añade elementos dinámicos durante el juego, como un indicador de vida o estado.	UI completa y bien integrada: menú inicial atractivo, elementos dinámicos en pantalla (puntuación, vida, etc.) y pantalla final (victoria o derrota) coherente con el juego.
<b>Programación 15%</b>	Implementa un script básico con una funcionalidad sencilla (por ejemplo, mover un objeto). Código sin estructura modular.	Implementa al menos 2 scripts funcionales con mecánicas básicas (por ejemplo, movimiento y una interacción simple). Código funcional pero con poca organización.	Implementa al menos 3 scripts bien estructurados y funcionales, utilizando métodos para organizar el código. Introduce una estructura lógica básica (por ejemplo, un temporizador).	Implementa 4 o más scripts con funcionalidades complejas (por ejemplo, sistemas de vida, habilidades o temporizadores avanzados). El código está modularizado con métodos reutilizables y comentarios claros.	Programación avanzada con 5 o más scripts que gestionen distintas partes del juego (movimiento, interacciones). Usa estructuras lógicas complejas y código bien organizado, modular y comentado, facilitando la legibilidad y el mantenimiento.
<b>Sonido 5%</b>		Añade al menos 2 efectos de sonido básicos (por ejemplo, pasos, saltos o colisiones). No incluye música de fondo.		Añade al menos 4 efectos de sonido coherentes con las acciones del juego (por ejemplo, pasos, ataques, recogida de objetos). Incluye una música de fondo funcional.	Implementa efectos de sonido bien sincronizados con las acciones principales (al menos 5). La música de fondo es adecuada para la temática del juego y puede incluir cambios dinámicos según el estado o progreso del jugador.
<b>Documentación 10%</b>		Ha entregado la documentación en los plazos establecidos.	La documentación es incompleta. Se incluyen algunas explicaciones básicas, pero falta claridad en la organización o en los detalles importantes del proyecto.	La documentación es clara y cubre los aspectos básicos del juego, como la mecánica principal, el diseño del escenario y los elementos importantes del código	La documentación es completa, detallada y profesional. Incluye una descripción exhaustiva del juego, las mecánicas, el diseño de los niveles, la programación y los problemas resueltos.