

«Talento Tech»

Desarrollo de Videojuegos

Unity 3D

Clase 14



Clase N° 14 | Partículas

Temario:

- Sistema de partículas.
 - Propiedades básicas del Sistema de Partículas.
 - Aplicaciones en VJ.
 - Ejemplos de uso.
-

Objetivos de la clase

Comprender el sistema de partículas en Unity.

- Explicar qué es el sistema de partículas y cómo funciona dentro del motor.
- Identificar los componentes principales de un Particle System en Unity.

Explorar las propiedades básicas del sistema de partículas.

- Configurar parámetros como tamaño, velocidad, dirección, vida útil y transparencia.
- Ajustar efectos visuales mediante módulos como Emission, Shape, Color over Lifetime y Noise.

Desarrollar ejemplos prácticos en Unity.

- Crear un sistema de partículas desde cero con efectos simples.
- Modificar y optimizar partículas para lograr el efecto deseado sin afectar el rendimiento.

¿Qué son los sistemas de partículas?

Los sistemas de partículas en Unity permiten simular efectos visuales dinámicos y realistas, como humo, fuego, lluvia, chispas y explosiones. A diferencia de los modelos 3D tradicionales, que son estáticos, las partículas están diseñadas para ser pequeñas imágenes (sprites) o mallas que se generan, se mueven y desaparecen con el tiempo. Unity proporciona el Particle System, una herramienta poderosa y versátil que permite configurar y personalizar estos efectos con gran detalle. Piensen en ella como una "fábrica" de elementos visuales en movimiento.

Usos en videojuegos

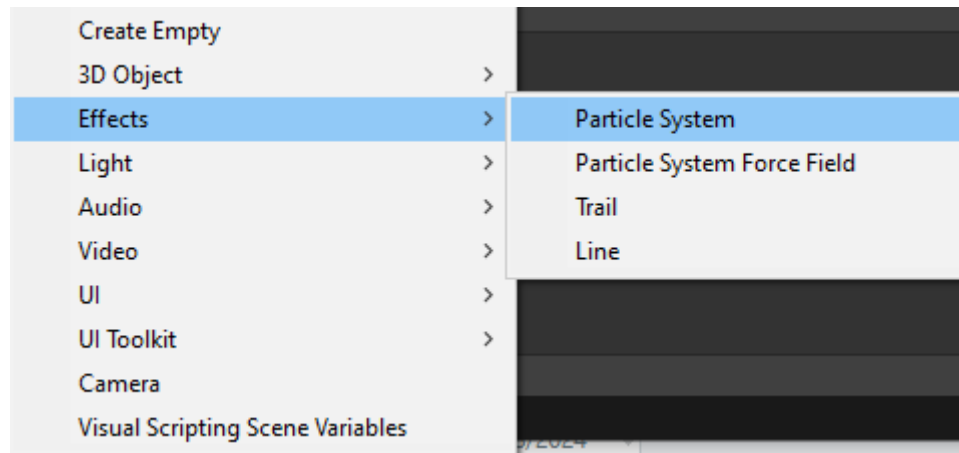
Los sistemas de partículas se utilizan en una variedad de situaciones para mejorar la inmersión y la estética de un juego. Al agregar estos efectos, se le da al jugador un feedback visual y se enriquece la experiencia. Algunos ejemplos comunes incluyen:

- **Fuego** 🔥: Simulación de antorchas, fogatas, o incluso un coche en llamas.
- **Humo** ☁️: Efectos de combustión, daño en vehículos, o escapes de vapor.
- **Explosiones** 💣: Impactos de proyectiles, granadas, o la destrucción de objetos.
- **Lluvia/Nieve** 🌧️❄️: Clima dinámico que ambienta las escenas.
- **Efectos mágicos** ✨: Hechizos, auras de personajes, escudos de energía, o partículas flotantes para dar ambiente.
- **Impactos de proyectiles** 💥: Balas golpeando superficies, chispas en metales.
- **Portales o teletransportación** 🌀: Animaciones de apariciones y desapariciones de personajes u objetos.

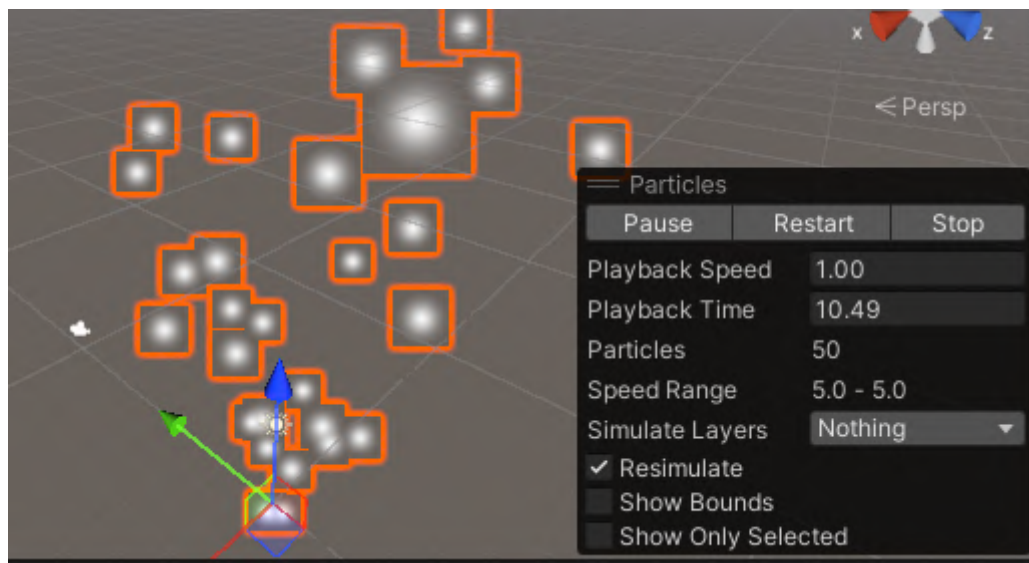
Estos efectos contribuyen significativamente a mejorar la experiencia del jugador al agregar realismo, impacto visual y retroalimentación a las acciones dentro del juego.

Bases del Sistema de Partículas

Primero iremos a la jerarquía ➡ Click Derecho ➡ Effects ➡ Particle System



Esto creará unas partículas básicas que se producirán para arriba y en un cuadro a la derecha nos empezará a mostrar opciones sobre su ciclo de vida.



Dentro del cuadro, encontraremos los siguientes controles que nos permiten ajustar sus efectos:

Playback Speed (Velocidad de Reproducción)

Este control les permite **acelerar o ralentizar la simulación** de sus partículas.

- Un valor **mayor a 1** va a hacer que el efecto se reproduzca más rápido.
- Un valor **menor a 1** lo va a ralentizar.

Esto es super útil para **observar el comportamiento** del efecto en detalle o ver cómo luce a diferentes ritmos.

Playback Time (Tiempo de Reproducción)

En esta opción van a ver el **tiempo actual de la simulación en segundos**. Lo mejor de todo es que pueden **modificar este valor** para avanzar o retroceder manualmente en la simulación. Es ideal para ir a un punto específico del efecto sin tener que esperar.

Particles (Cantidad de Partículas)

Este indicador les muestra la **cantidad actual de partículas activas** en su escena en ese preciso momento. Es una métrica clave para **monitorear el rendimiento**. Si ven un número muy alto, podría ser una señal de que el sistema está generando demasiadas partículas y podría afectar la fluidez de su juego.

Speed Range (Rango de Velocidad)




Muestra el **rango de velocidades** de las partículas que están activas. Es decir, les indica cuál es la partícula más lenta y la más rápida en ese instante. Esto nos permite **verificar la variabilidad de la velocidad** entre las partículas y asegurarse de que el efecto se comporta como se espera.

Resimulate (Casilla de Verificación)

Si activan esta casilla, **cualquier cambio que hagan en los parámetros** del sistema de partículas se va a aplicar **en tiempo real**. No van a necesitar reiniciar la simulación cada vez que ajusten algo. Esto nos **ahorará muchísimo tiempo** al afinar sus efectos.

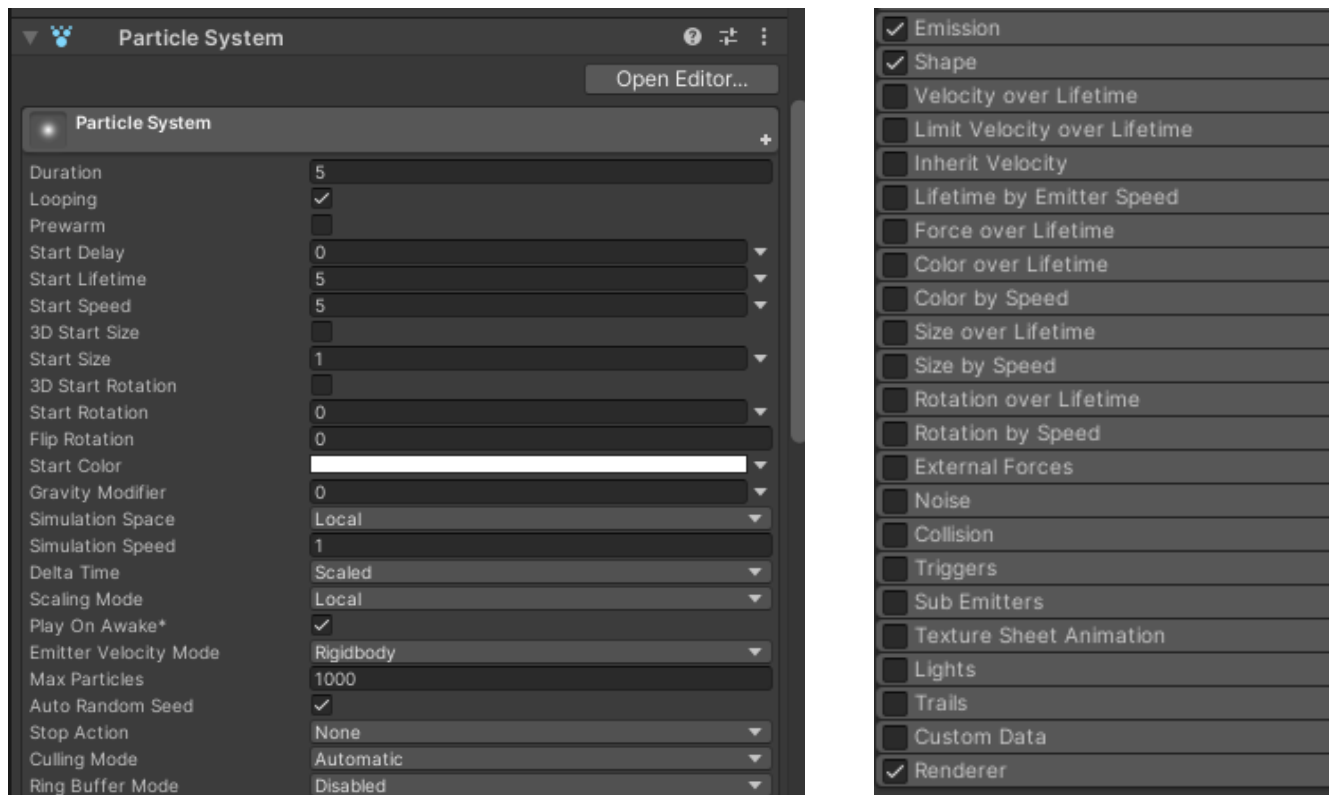
Playback Controls (Botones de Play, Pause, Stop)

Estos son sus controles básicos para manejar la simulación:

- **Play** : Inicia o reanuda la simulación del sistema de partículas.
- **Pause** : Pausa la simulación en el estado actual, permitiéndolo inspeccionar las partículas en un momento específico.
- **Stop** : Detiene la simulación y **reinicia el sistema** a su estado inicial.

Inspector y variables

Si vemos el Inspector al seleccionar el Sistema de Partículas, veremos lo siguiente:



Al darle click en Particle System vamos a modificar las siguientes opciones:

Main

Dentro del main nos encontramos con las variables principales del sistema.

1. **Duration (Duración)** Define el **tiempo total** que el sistema de partículas estará activo, en segundos. Si su efecto no está en modo "Looping", se detendrá completamente después de este tiempo

Ejemplo: Si ponen Duration en **5**, las partículas se reproducirán por 5 segundos y luego se detendrán.

2. **Looping (Bucle)** Activa o desactiva la **repetición automática** del sistema de partículas. Si está activado, el efecto se reiniciará automáticamente una vez que termine su "Duration". Es ideal para fuego constante o lluvia sin fin.

3. **Prewarm (Precalentar)** *Solo funciona si "Looping" está activado.* Esta opción permite que el sistema comience **como si ya hubiera estado funcionando** por un tiempo. Es decir, no verán el inicio "desde cero" del efecto, sino un estado ya avanzado.

Útil para: Efectos como una fogata o lluvia que necesitan verse activos y completos desde el primer momento en que el juego carga la escena.

4. **Start Delay (Retraso de Inicio)** Especifica un **tiempo de espera** antes de que las partículas empiecen a emitirse. Pueden definir un valor fijo o un rango aleatorio para más variedad.

Ejemplo: Si configuran Start Delay en 2, las partículas no comenzarán a generarse hasta 2 segundos después de que el sistema se active.

5. **Start Lifetime (Vida Útil Inicial)** Determina **cuánto tiempo vive cada partícula individual** antes de desaparecer. Pueden darle un valor fijo o un rango aleatorio para un comportamiento más natural.

Ejemplo: Con Start Lifetime en 3, cada partícula se extinguirá después de 3 segundos de ser creada.

6. **Start Speed (Velocidad Inicial)** Define la **velocidad con la que nacen** las partículas. Como en la vida útil, pueden usar un valor fijo o un rango aleatorio.

Ejemplo: Si ponen Start Speed en 5, las partículas saldrán disparadas rápidamente desde su origen.

7. **3D Start Size (Tamaño Inicial 3D)**

- **Si está desactivado**, todas las partículas tendrán el mismo tamaño en los tres ejes (X, Y, Z).
- **Si está activado**, podrán establecer tamaños diferentes para cada eje, lo que es genial para partículas que tienen una forma específica o que quieren estirar en alguna dirección

8. **Start Size (Tamaño Inicial)** Aquí definen el **tamaño inicial** de las partículas. Pueden establecer un valor fijo o un rango aleatorio para que cada partícula tenga un tamaño ligeramente diferente.

Ejemplo: Si Start Size es 2, las partículas serán el doble de su tamaño base.

9. **3D Start Rotation (Rotación Inicial 3D)** Si activan esto, pueden aplicar **rotaciones diferentes en los ejes X, Y y Z** a las partículas al nacer. Si no está activado, la rotación será uniforme en los tres ejes

10. **Start Rotation (Rotación Inicial)** Define la **rotación inicial** de las partículas en grados. Al igual que otras propiedades, pueden usar un valor fijo o un rango aleatorio para añadir variedad.

11. **Flip Rotation (Invertir Rotación)** Invierte la rotación de un porcentaje de las partículas (de 0 a 1, donde 1 es 100%). Esto es muy **útil para variar la apariencia** de las partículas, especialmente cuando usan sprites o texturas que tienen una dirección.

12. **Start Color (Color Inicial)** Define el **color inicial** de las partículas. Pueden elegir un color fijo o un rango de colores para que las partículas nazcan con tonalidades diferentes.

13. **Gravity Modifier (Modificador de Gravedad)** Ajusta **cuánto le afecta la gravedad** a las partículas.

- 0: No hay efecto de gravedad (las partículas no caen).
- 1: Usa la gravedad estándar de Unity.
- **Valores negativos:** Harán que las partículas "floten" o se muevan hacia arriba, ¡ideal para humo o burbujas!

14. **Simulation Space (Espacio de Simulación)** Define en qué "espacio" se mueven las partículas:

- **Local:** Las partículas se mueven **junto con el objeto** que emite el sistema. Si el objeto se mueve, las partículas se arrastran con él.
- **World:** Las partículas se mueven **libremente en el mundo**, independientemente del movimiento del objeto emisor.

- **Custom:** Permite usar otro objeto como referencia para el espacio de simulación.

15. Simulation Speed (Velocidad de Simulación) Modifica la **velocidad global de todo el sistema de partículas**.

- Un valor **mayor a 1** acelera todo el efecto.
- Un valor **menor a 1** lo ralentiza.

16. Delta Time (Tiempo Delta) Define **cómo se calculan los pasos de tiempo** de la simulación:

- **Scaled (por defecto):** La simulación usa el tiempo normal del juego y se ve **afectada por `Time.timeScale`**. Si pausan el juego, las partículas también se pausan.
- **Unscaled:** La simulación **no se ve afectada por `Time.timeScale`**. ¡Útil para partículas que deben seguir activas incluso si el juego está en pausa!

17. Scaling Mode (Modo de Escala) Define **cómo se escala** el sistema de partículas cuando el objeto emisor cambia de tamaño:

- **Hierarchy:** El sistema se escala **junto con el objeto padre** en la jerarquía.
- **Local:** Solo afecta el **tamaño individual de las partículas**, sin modificar su velocidad.
- **Shape:** La escala se basa en la **forma del emisor**

18. Play On Awake (Reproducir al Iniciar)

- Si está **activado**, el sistema de partículas se **reproduce automáticamente** apenas la escena comienza.
- Si está **desactivado**, van a necesitar activarlo manualmente (por ejemplo, desde un script o usando el método `Play()`)

19. Emitter Velocity Mode (Modo de Velocidad del Emisor) Controla cómo las partículas **heredan la velocidad del objeto que las emite**:

- **Off:** Las partículas se generan **sin tomar en cuenta** la velocidad del emisor.
- **Rigidbody:** La velocidad de las partículas se ve afectada por la velocidad del **Rigidbody** del emisor.

- **Transform:** La velocidad de las partículas se ve afectada por el **movimiento del objeto Transform** del emisor.

20. Max Particles (Máximo de Partículas) Establece el **número máximo de partículas** que el sistema puede generar y tener activas al mismo tiempo. Limitar este valor es **crucial para la optimización del rendimiento** y evitar que su juego se ralentice

21. Auto Random Seed (Semilla Aleatoria Automática)

- Si está **activado**, Unity generará un valor aleatorio diferente para la simulación **cada vez que se ejecute**, resultando en un comportamiento ligeramente distinto.
- Si está **desactivado**, pueden fijar un "Random Seed" (una semilla) para que el comportamiento del efecto sea **siempre el mismo** cada vez que se reproduzca, lo cual es útil para depurar o si necesitan consistencia absoluta.

Creando nuestras Partículas:

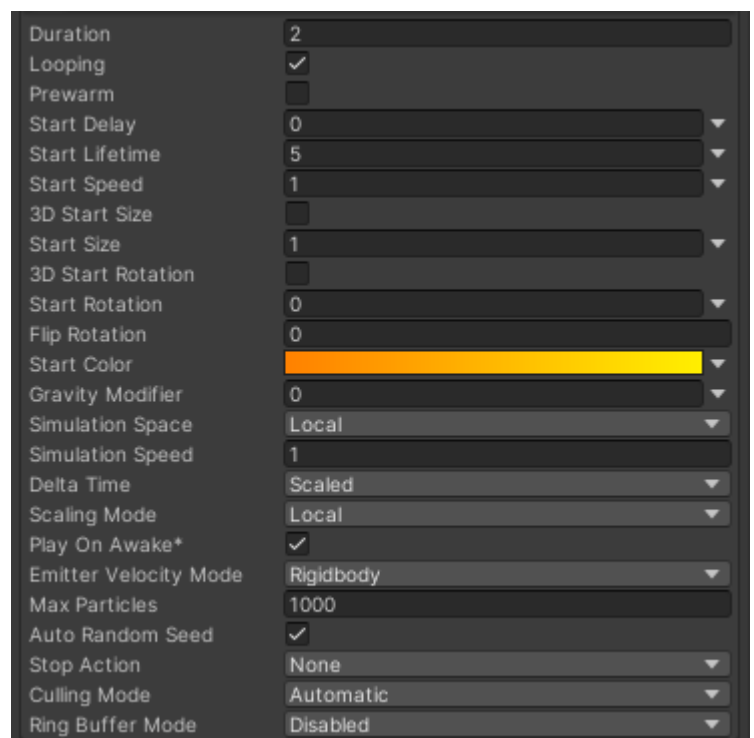
🔥 Fuego:

1. Crear el sistema de partículas:

- En el **Hierarchy**, haz clic derecho → *Effects* → *Particle System*.
- Renómbralo a **"Fire"** para mantener el orden.

2. Configuración del módulo Main:

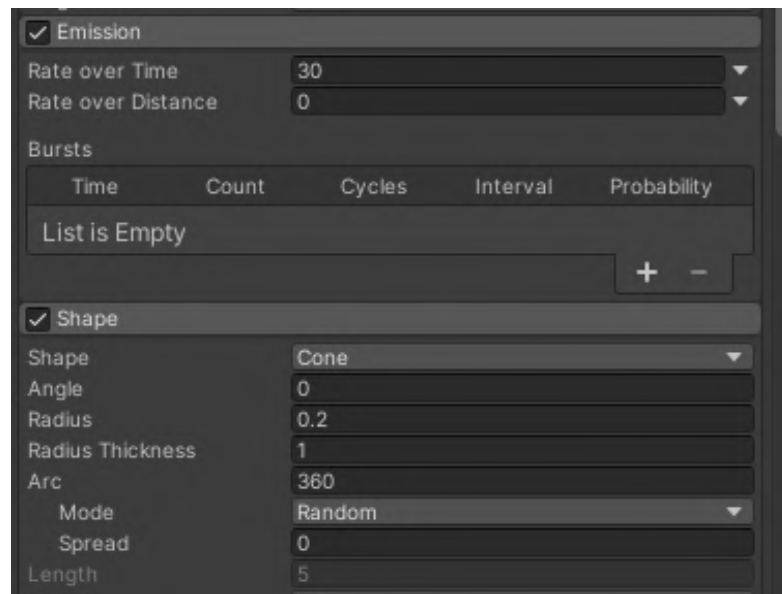
- **Duration:** 2
- **Looping:** ☒ (Activado)
- **Start Lifetime:** Entre 1 - 2 (Rango aleatorio)
- **Start Speed:** Entre 0.5 - 1
- **Start Size:** Entre 0.5 - 1
- **Start Color:** Gradiente de naranja a amarillo 🔥
- **Gravity Modifier:** 0



3. Clickeamos en Emisión y Forma:

Ahora, le diremos a nuestro sistema cómo y dónde emitir esas partículas:

- **Emission:**
 - **Rate over Time:** 30 (Esto significa que se generarán 30 partículas por segundo, creando un flujo constante de llamas).
- **Shape:**
 - **Tipo: Cone.** Un cono es perfecto para la forma ascendente de una llama.
 - **Angle: 0** .Para que las partículas salgan rectas hacia arriba, simulando el centro de la llama.
 - **Radius: 0.2** .Define el ancho de la base del cono de emisión.



4. Añadir Color Over Lifetime:

Para que el fuego parezca más realista, queremos que su color y tamaño cambien a medida que las partículas envejecen:

Añadir Color Over Lifetime:

- Activa el módulo **Color over Lifetime**.
- Usa un gradiente: **naranja** → **amarillo** → **transparente**. Esto hará que las llamas se desvanezcan suavemente.

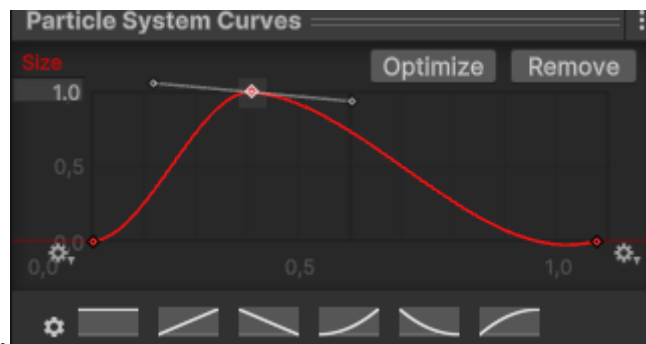
Añadir Size Over Lifetime:

Activa el módulo **Size over Lifetime**



Configura el gráfico para que las partículas **crezcan ligeramente al principio** y luego **se encojan** hasta desaparecer.

Al darle click a la línea de **Size**, vamos a poder modificar como queremos que crezcan las partículas. Si le damos click izquierdo vamos a poder agregar mas puntos de quiebre a esa línea



5. **Añadiendo Dinamismo (Noise - opcional):**

- Activa el módulo **Noise** para darle más dinamismo.
- Strength: 0.2. Esto añadirá pequeñas variaciones al movimiento de las partículas, haciendo que las llamas "bailen" un poco



6. **Textura y Material para el fuego:**

Un sistema de partículas necesita un Material y una Textura para verse bien. Aquí vamos a crear un material específico para el fuego.

a. **Obtener la textura de fuego:**

Pueden encontrar la textura en este link:

<https://drive.google.com/drive/folders/19TMho3Nn1q-Z4PWzmx07XzU06bbL-R1q?usp=sharing>

Crea un **material** con un shader de partículas y usa una textura de fuego.

b. Editar la textura:

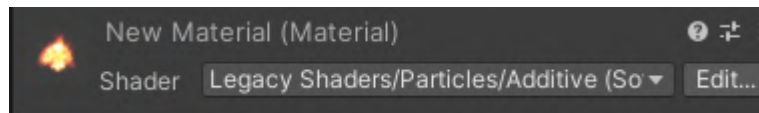
Arrastra la imagen de la textura descargada a la carpeta Assets en Unity.

Selecciona la textura y, en el **Inspector**, cambia estas configuraciones que nos asegurarán que la textura tenga buena calidad y un fondo transparente:

- **Texture Type:** Default
- **Alpha Is Transparency:** ☒ (Activado)
- **Wrap Mode:** Repeat
- **Filter Mode:** Bilinear
- **Compression:** None

c. Crear un Material para las Partículas

- Ve a la carpeta **Assets**, haz clic derecho → *Create* → *Material*.
- En el **Inspector**, cambia el **Shader** a **Additive (Soft)**



d. Aplicar la Textura al Material

- En el **Inspector** del material, busca la opción **Albedo / Main Texture**.
- Arrastra la textura de fuego a este campo.
- Ajusta los valores de **Tint Color** para que el fuego tenga el color deseado.

e. Asignar el Material al Sistema de Partículas

- Selecciona el sistema de partículas que creaste.
- En el **Inspector**, busca el módulo **Renderer**.
- En la opción **Material**, arrastra **"Fire_Material"**.

Render: Cómo se muestran nuestras partículas

Renderizar es el proceso de generar la imagen final de un objeto 3D o 2D en la pantalla. La computadora toma la información del objeto (forma, textura, color, iluminación) y la convierte en **píxeles visibles** en la escena del juego. Este proceso lo realiza la **GPU (Tarjeta Gráfica)**, y puede ser más rápido o más lento dependiendo de la cantidad de detalles a mostrar.

Entonces un **Renderer** es un componente en Unity que le **permite a un objeto aparecer en la pantalla**.

El **Renderer** le dice a Unity **cómo dibujar un objeto**, tomando en cuenta su forma (malla), materiales, texturas y efectos visuales.

Que nuestro sistema de partículas tenga una opción de "Render" es vital. Esta nos dará la capacidad de convertir casi cualquier cosa en partículas. Veamos sus opciones a destacar:

1. Material

Define la textura o shader que se aplicará a las partículas.



Opciones Claves:

- Puedes asignar un Material con una textura para cambiar el aspecto de las partículas.
- Se recomienda usar shaders específicos de partículas como:
 - **Particles/Unlit** → No se ve afectado por la luz. Útil para efectos que siempre deben brillar igual
 - **Particles/Additive** → Hace que las partículas brillen y se mezclen con el fondo. Ideal para fuego o magia.
 - **Particles/Alpha Blended** → Usa transparencia para suavizar los bordes.

Ejemplo: Para hacer fuego, asigna un material con una textura de llamas y usa el shader Particles/Additive.

2. Render Mode

Cambia la forma en que se representan las partículas en la escena.



Opciones Claves:

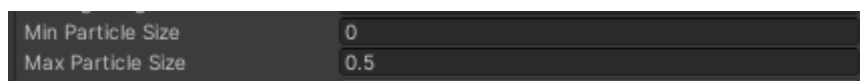
1. **Billboard** (Por defecto) → Las partículas siempre miran a la cámara. 🔄 Es el modo más común y eficiente para la mayoría de los efectos (humo, fuego, chispas)
2. **Horizontal Billboard** → Las partículas siempre están planas en el suelo.
3. **Vertical Billboard** → Las partículas siempre son verticales pero giran hacia la cámara.
4. **Mesh** → En lugar de usar un sprite, se usa un modelo 3D para cada partícula.

Ejemplo:

- Billboard es ideal para humo, fuego y efectos mágicos.
- Mesh es útil para proyectiles 3D como flechas o esferas de energía.

3. Min Particle Size / Max Particle Size

Controla el tamaño mínimo y máximo que pueden tener las partículas en pantalla.



Opciones Claves:

- Si Max Particle Size es muy grande, las partículas podrían cubrir demasiado espacio en la pantalla.
- Si Min Particle Size es muy pequeño, podrían no ser visibles.

Ejemplo: Si creas partículas de polvo en un juego 3D, puedes limitar su tamaño para que no crezcan demasiado en la distancia.

4. Flip

Invierte las partículas en X o Y, dándoles variedad visual sin necesidad de nuevas texturas.



Opciones Claves:

- **Flip X:** Invierte la partícula horizontalmente.
- **Flip Y:** Invierte la partícula verticalmente.

Ejemplo: Si tienes partículas de hojas cayendo, activar Flip X y Flip Y puede hacer que cada hoja tenga una orientación diferente, creando un efecto más natural. 🍂

Conclusión

El sistema de partículas es bastante complejo, su gran número de herramientas y versatilidad para mezclarlas permite generar millones de posibilidades distintas. En la clase pudimos ver algunas de las más importantes. Les sugerimos jugar con ellas y crear sus propias ideas.



Efectos Especiales:

*Después de semanas de arduo trabajo en **Nexus**, el equipo de **TalentoLab** ha demostrado ser una pieza clave en el desarrollo del proyecto. Las mecánicas básicas, el sistema de enemigos y la optimización del rendimiento están avanzando a paso firme.*

Sin embargo, los diseñadores y artistas han señalado algo importante: el mundo de Nexus se siente vacío.

"El motor gráfico ya es eficiente, pero nos falta algo más... ¡Necesitamos vida, dinamismo! Fuego ardiendo en las antorchas, humo saliendo de los respiraderos, chispas en los impactos, y energía fluyendo por los portales. Todo esto hará que el mundo cobre vida y se sienta realmente inmersivo."

*Para lograr esto, la **nueva misión** es dominar el **Sistema de Partículas de Unity**.*

Ejercicios prácticos:

Creando Efectos Únicos para Nexus

El equipo de **TalentoLab** ha dado los primeros pasos en la implementación del **Sistema de Partículas**, logrando efectos básicos como fuego y humo. Durante una reunión con el equipo de dirección artística de **Nexus**, surge una nueva solicitud:

"Necesitamos efectos visuales que sean únicos y memorables. Algo que haga que nuestro mundo se sienta especial y diferente de cualquier otro juego."



Luigi cree que es el momento ideal para que aportes tu creatividad y dejes un sello en el proyecto.

Objetivo: Experimentar con el **Particle System** de Unity y crear efectos únicos para el mundo de Nexus.

1. **Explorar las configuraciones del sistema de partículas**, modificando color, tamaño, velocidad, colisiones y efectos especiales.

2. **Diseñar un efecto original**, ya sea un aura mágica, una explosión distinta, una neblina sobrenatural o cualquier otra idea innovadora.
3. **Justificar cómo este efecto podría encajar en el mundo del juego**, pensando en qué mecánica, entorno o narrativa lo integraría mejor.

Materiales y recursos adicionales.

Introducción a Sistema de Partículas

<https://learn.unity.com/tutorial/introduction-to-particle-systems>

Documentación

<https://docs.unity3d.com/2022.3/Documentation/Manual/ParticleSystems.html>

Preguntas para reflexionar.

1. ¿Qué partículas sería interesante crear?
 2. ¿Podríamos usarlas para indicar la naturaleza de la situación u objeto?
-

Próximos pasos.

En la próxima clase veremos una introducción a juegos Mobile. Pasaremos a adaptar nuestros controles básicos para poder usarlos en Celular y crearemos una Build para jugarlo



Buenos Aires
aprende
Agencia de Habilidades para el Futuro

BA Buenos
Aires
Ciudad