

Sonido en videojuegos

FMOD Studio I

El ejercicio marcado con **Evaluable** debe entregarse a través del CV. Se subirá únicamente un archivo **NombreApellidos1-NombreApellidos2.zip** (con los nombres de los alumnos del grupo) que con un proyecto FMOD que contendrá el evento pedido. No subir el proyecto compilado! (la carpeta build con los bancos generados).

El ejercicio marcado con **Evaluable** debe entregarse a través del CV. Para realizarlo:

- Generarse un nuevo proyecto FMOD.
- Incorporar los assets necesarios e implementar el evento pedido.
- Comprimir el proyecto generado en un archivo **NombreApellidos1-NombreApellidos2.zip** (con los nombres de los alumnos del grupo). Solo subirá el archivo uno de los miembros del grupo.
- **No** incluir el proyecto compilado! (la carpeta build con los bancos generados). Es necesario poder abrir el proyecto en el editor de FMOD.

1. (**Evaluable**) En este ejercicio vamos a crear un evento de sonido de pasos en el editor FMOD Studio. En primer lugar creamos un evento 3D Timeline *pasos*, añadimos las 3 muestras *steps* como assets y las cargamos en el track como multi-instrumento (arrastrar simultáneamente al track desde la pestaña de assets). Probaremos el correcto funcionamiento al reproducir las mismas. A continuación vamos a enriquecer el evento como sigue:

- Aleatorizar elección de cada una de las muestras para introducir variación sonora en cada paso. Utilizar los porcentajes 40%, 30%, 30% para las muestrass.
- Para darle aún más riqueza, sobre el pitch del instrumento (no sobre el pitch individual de cada muestra) añadiremos una modulación aleatoria de ± 2 semitonos de variación.
- A continuación crearemos un parámetro *Recinto* para añadir al sonido una cantidad variable de reverb en función del valor de dicho parámetro. Para ello:
 - Sobre el master track del evento creado añadir el efecto *FMOD Reverb* en la pestaña *Pre* (antes del fader).
 - Para esta reverb, sobre el control *Wet Level* crear una pista de *Automatización* (botón derecho del ratón sobre la rueda de control).
 - A continuación, sobre esta automatización (botón *Add Curve*) crear el nuevo parámetro *Recinto* de tipo *Continuous* con rango de variación [0,1]. Definir el nivel *Wet Level* de $-\infty$ dB a 0dB. Esta curva no debe definirse en el track de automatización (ligado al timeline), sino en el Deck, para que se ligue al valor del parámetro.
- Ahora vamos a introducir un nuevo parámetro *velocidad* para controlar la frecuencia de los pasos (sin alterar el pitch). Sobre el parámetro *Timeline* vamos a definir *etiquetas condicionales de retorno*:
 - En la región de definición de etiquetas (región negra horizontal sobre el track) con el botón derecho, añadir un marcador al principio de las muestras. A continuación añadir una etiqueta de transición hacia el marcador anterior tras el punto de terminación de las muestras. Obsérvese que cuanto más se retrasa este punto de retorno más lentos se hacen los pasos.
 - Añadir el nuevo parámetro *velocidad* de tipo *Labeled* y valores *Lento*, *Medio*, *Rápido*.
 - Sobre la etiqueta de retorno anterior, añadir una condición ligada al parámetro velocidad de modo que se aplique la etiqueta para un intervalo definido de dicho parámetro.
 - Añadir más etiquetas de retorno para los distintos valores del parámetro velocidad con retraso temporal adecuado a cada una de ellas, en función del parámetro velocidad.

2. La solución planteada tiene algunos problemas. Para empezar, las etiquetas de retorno solo nos permiten tener valores "discretos" de velocidad (no podemos tener un número finito de dichas etiquetas). Por otro lado, al incluir la reverb, se alarga la duración de los sonidos y el multi-instrumento solo reinicia la reproducción cuando acaba la reproducción del sonido (incluida la reverb). Una forma de solucionar estos problemas es:

- Crear un parámetro velocidad.
- Cargar las muestras en un *scatterer instrument* en un track en dicho parámetro.
- Crear una region de loop para este instrumento
- Ajustar polifonía a 1.
- Ajustar *Min & Max Spawn interval* a la duración de la muestra (o ligeramente por debajo).
- Automatizar *Spawn Rate* para que se incremente con la velocidad y controlar con parámetro *velocidad*.

Por último crear un banco de sonido y compilar el banco (build). Utilizando el API de FMOD Studio, hacer un pequeño programa para cargar el banco, y reproducir el evento variando los parámetros que hemos creado. Para ello puede utilizarse como punto de partida alguno de los ejemplos incluidos en la distribución de FMOD (por ejemplo, *simple_event.cpp*).