

Estructura del Código en Phaser

Ahora es momento de aprender la estructura de archivos y de código que tiene la plantilla de Phaser.

Los archivos “rollup.config.dev.js” y “rollup.config.dist.js” contienen información sobre los plugins y la ejecución del proyecto en el navegador, como el puerto en el que se aloja y el comportamiento de dichos plugins.

El archivo principal del juego se encuentra en src y se llama “game.ts”. Tiene primero los imports necesarios y comienza declarando una clase que hereda de Phaser.Scene.

Su constructor llama al constructor de su clase padre.

Contiene un objeto de configuración que es una constante, donde :

- Definimos el tipo de contenedor que va a tener el juego (Canvas o WebGL) que está como AUTO, con lo que definirá el mejor contenedor posible.
- El color de fondo.
- El ancho y el alto de la escena.
- La escena inicial del juego.

Después de este objeto hay otra constante con un objeto Phaser.Game, que inicia el proceso de arranque de la escena pasándole la constante de configuración como parámetro.

Una escena de Phaser tiene 4 métodos predefinidos :

- init : Donde se inicializan las variables
- preload : Donde se precargarán todos los assets
- create : Donde se creará el mundo y se compondrá la escena
- upload : Se ejecutará constantemente ya que lo hace cada frame

Tienen un parecido muy similar a las funciones que tienen los objetos Screen de LibGDX.

Ahora, eliminamos los archivos que lleva la plantilla de base y no nos sirven para hacer nuestro propio juego. Y así podemos proceder a limpiar el código de game.ts.