


Tema	CLASE CAPSTONE: EL JUEGO MÁS DIFÍCIL DEL MUNDO	
Descripción de la clase	El alumno usará su conocimiento de sprites, funciones, bucles y sonido para crear el juego más difícil del mundo.	
Clase	C8	
Duración de la clase	45 minutos	
Objetivo	<ul style="list-style-type: none"> • Construir "El juego más difícil del mundo". • Repasar los conceptos de las clases anteriores. 	
Recursos requeridos	<ul style="list-style-type: none"> • Recursos para maestras <ul style="list-style-type: none"> o Iniciar sesión en code.org. o Laptop con conectividad a internet. o Auriculares con micrófono. o Cuaderno y bolígrafo. • Recursos para alumnos <ul style="list-style-type: none"> o Iniciar sesión en code.org. o Laptop con conectividad a internet. o Auriculares con micrófono. o Cuaderno y bolígrafo. 	
Estructura de la clase	Diapositivas de Rompiendo el hielo Actividad dirigida por la maestra Actividad dirigida por el alumno Conclusión – opción con diapositivas	15 Mins 8 Mins 30 Mins 5 Mins
<u>ROMPIENDO EL HIELO - 15 minutos</u>		
<div>  </div> <p>La maestra inicia la presentación desde la diapositiva 1 a la 15 Consulta las notas en tu pantalla y sigue las instrucciones de cada diapositiva.</p>		
Acción de la maestra		Acción del alumno

Hola (nombre del alumno). ¿Cómo estás? ¡Qué gusto verte! Antes de comenzar con esta emocionante clase, ¿recuerdas qué es lo que vamos a aprender hoy?

Ejecuta la presentación de la diapositiva 1 a la 10.

Esto es lo que harás en la sesión Rompiendo el hielo:


- Explica la importancia de la clase capstone.
- Recuerda el aprendizaje de la clase anterior.
- Ayuda al alumno a recordar los conceptos vistos hasta ahora para diseñar un juego completo.


REA: Hola, gracias, ¡sí!

El alumno recuerda lo que la maestro mencionó que se cubriría en esta clase la clase la sesión anterior.

Haz clic en la pestaña de diapositivas y presenta las diapositivas.

Sesión PyR

Pregunta	Respuesta
<p>Selecciona la línea de código que movería la paleta de la computadora de manera vertical con la pelota.</p>  <p>A. computerPaddle.y = ball.y; B. computerPaddle.x = ball.x; C. computerPaddle.y = World.mouseY; D. computerPaddle.y = ball.Y;</p>	A
<p>Selecciona la instrucción para hacer que la pelota rebote en la playerPaddle - <i>paleta del jugador</i>.</p> <p>A. ball.isTouching(playerPaddle); B. ball.collide(playerPaddle); C. ball.bounce(playerPaddle); D. ball.bounceOff(playerPaddle);</p>	D

Detalles de la actividad	Solución/Lineamientos
<p>Ejecuta la presentación de la diapositiva 11 a la 15 para establecer el enunciado del problema</p> <p>Haz lo siguiente en esta sesión:</p> <ul style="list-style-type: none"> • Usa pistas para enseñar a los alumnos como enfrentar cualquier problema complejo. • Explica “el juego más difícil del mundo”, que necesita ser diseñado en la clase de hoy. 	<p>Narra la historia usando gestos con las manos y métodos de modulación de voz, para atraer más interés del alumno.</p>
<p>La maestra finaliza la presentación de diapositivas</p> 	
<p>ACTIVIDAD DIRIGIDA POR LA MAESTRA - 8 minutos</p>	
<p>La maestra comienza a compartir su pantalla</p>	
<p><u>DESAFÍO</u></p> <ul style="list-style-type: none"> • Deconstruye “El juego más difícil del mundo” • Pide al alumno que recuerde los conceptos que pueden ser usados para crear el juego. 	
Acción de la maestra	Acción del alumno
<p>Paso 2:</p> <p>Actividad dirigida por la maestra</p> <p><i>La maestra abre el juego para hablar de la Actividad dirigida por la maestra 1</i></p> <p>¿Te gustó jugar el juego?</p> <p>¿Qué haría al juego más difícil?</p>	<p>REA:</p> <ul style="list-style-type: none"> - Más cuadros rojos. - Mayor velocidad de los cuadros rojos.
<p>En la clase de hoy, construiremos este juego mientras revisamos todos los conceptos que aprendimos en la clase anterior.</p> <p>Al completar esta clase, tendrás “El juego más difícil del mundo” con el cual puedes retar a tus amigos para que lo jueguen. ¿Cómo te sientes al respecto</p>	<p>REA:</p> <p>variadas</p>

<p>Si recuerdas, en el juego de escape, deconstruimos el juego en pequeños componentes. Mover la plataforma, pelota que rebota, creación de ladrillos, colisión con los ladrillos. A esto se le llama deconstrucción. Antes de trabajar en un proyecto, un programador deconstruye la tarea difícil en pequeñas y más sencillas tareas. Cada tarea debe ser lo bastante simple para pensar en cómo abordarla.</p> <p>¿Cómo dividirías este juego en componentes más simples?</p>	<p>REA:</p> <ul style="list-style-type: none"> - Las paredes del túnel en el cual el juego se juega el juego. - Cuadros rojos rebotando en las paredes - El cuadro verde que puede ser controlado con las flechas derecha e izquierda. - Restablecer el juego cuando el cuadro toque los cuadros verdes. - Conteo de muertes - Agregar sonidos
<p>Increíble. Opcionalmente, también podemos agregar más niveles al juego incrementando el reto para el usuario. ¿Cómo agregamos más niveles?</p>	<p>REA: Usando estados del juego</p>
<p>Antes de que empecemos a programar el juego más difícil del mundo, aprendamos cómo mover un sprite presionando una tecla.</p> <p>En juegos debes haber usado el teclado para controlar el movimiento del jugador.. Nosotros también moveremos nuestro sprite basado en qué tecla de flecha se presione.</p> <p>Tenemos un sprite de caja en nuestro canvas.</p> <p><i>La maestra abre la Actividad de la maestra 3.</i></p> <p><i>La maestra escribe el código.</i></p>	

```
var box = createSprite(200,200,20,20);  
  
function draw()  
{  
  background("white");  
  drawSprites()  
}
```



Queremos mover este sprite de caja hacia el lado derecho cuando el usuario presione la tecla de la flecha derecha. Para eso utilizaremos la función **keydown()**. Esta función tomará el nombre clave como un argumento. Por ejemplo, si queremos detectar la tecla de flecha izquierda entonces pasaremos "left" -*izquierda* como un argumento. La función se convertirá en **keydown("left")**.

En la función draw escribiremos el código para detectar la tecla presionada y una vez que esté hecho, ejecutaremos el código para mover el sprite en la dirección deseada.

Primero, escribiremos una condición para revisar cual está siendo presionada por el usuario.
En el bloque de condicional if, escribiremos el código para mover el sprite de la caja.

```
var box = createSprite(200,200,20,20);

function draw()
{
  background("white");
  if(keyDown("right"))
  {
    // Mover el sprite a la derecha
  }
  drawSprites()
}
```

¿Puedes decirme cómo podemos mover el sprite?

Para mover un sprite tomaremos su posición, ahora basado en el hecho de la dirección que quieras mover el sprite. Agregamos o restamos un valor desde su posición actual. En nuestro caso, queremos movernos a la derecha y sabemos que en nuestro canvas la posición x cambia en posición horizontal y la posición y cambia en posición vertical.

Pero, ¿qué hacemos, incrementamos la posición de x o disminuimos el valor de la posición de x?

Cuando nos movemos de izquierda a derecha nuestro valor de posición x incrementa.

Por ejemplo, si quieres crear un sprite en la esquina izquierda del canvas, ¿cuál sería la posición de x para eso?

Mantendremos el valor de x más bajo

createSprite(50,100,20,20).

REA:

Variada.

REA:

Variada.

REA:

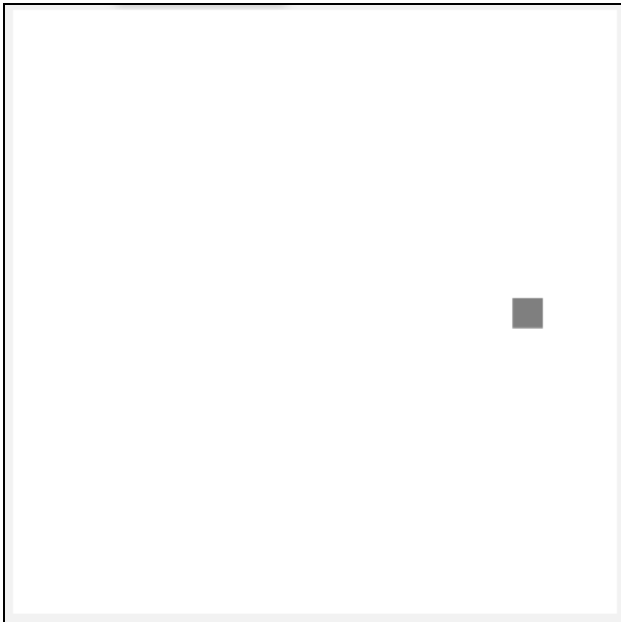
Variada.

Del mismo modo, si queremos crear un sprite en la esquina derecho usaremos un valor más alto de la x
createSprite(300,100,20,20)

Esto significa que si queremos mover la caja hacia la derecha tenemos que incrementar la posición x. Eso lo podemos hacer agregándole un número. Entre más alto sea el número más será el movimiento del sprite.

```
var box = createSprite(200,200,20,20);

function draw()
{
  background("white");
  if(keyDown("right"))
  {
    box.x = box.x +3;
  }
  drawSprites()
}
```



Si quieres que el sprite se mueva hacia la izquierda entonces tenemos que escribir otra condición como:
if (keydown("left"))

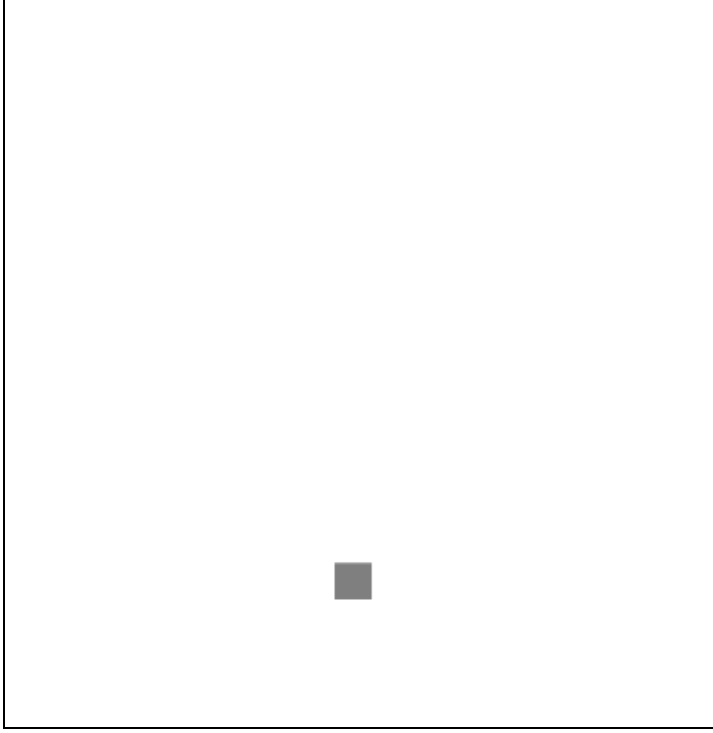

```
{  
box.x = box.x-3;  
}
```


Esto moverá la caja hacia la izquierda cuando el usuario presione la tecla de la flecha izquierda.

Lo mismo pasa con el movimiento hacia arriba y abajo, si queremos mover la caja hacia abajo tenemos que incrementar la posición de y.

Para mover hacia arriba, disminuirémos la posición y.

```
var box = createSprite(200,200,20,20);  
  
function draw()  
{  
  background("white");  
  if(keyDown("right"))  
  {  
    box.x = box.x +3;  
  }  
  
  if(keyDown("down"))  
  {  
    box.y = box.y +3;  
  }  
  drawSprites()  
}
```


	
<p>Hemos aprendido cómo mover el sprite al basado en la tecla que el usuario presione.</p> <p>Ahora comencemos a programar cada pequeño componente del juego. Observarás como el juego entero se construye resolviendo pequeños componentes en el juego.</p> <p>Comencemos.</p>	-
<div style="text-align: center;">  <p>La maestra inicia la presentación de diapositivas desde la diapositiva 16 a la 17 (solo 1 diapositiva para esta actividad)</p> </div>	
<p>Ejecuta la presentación con la diapositiva 16 para establecer el contexto de la actividad del alumno.</p>	
<p>Aquí hay un reto para ti, tienes que diseñar el juego más difícil del mundo.</p> <p>Ten presente todos los conceptos que has aprendido hasta ahora en cuanto a objetos de sprites, propiedades,</p>	<p>REA:</p> <p>Empezaremos por</p> <ul style="list-style-type: none"> - crear 12 - 1 ding - 4 dongs

funciones, enunciados condicionales, estados del juego, etc. ¿Puedes decirme cuales son los varios pasos involucrados y cómo empezarías?	<ul style="list-style-type: none"> - asignar velocidad Y a los dongs - usar condición if para mover el sprite de ding - usar el estado del juego - agregar efectos de sonido al juego - calificar al jugador
¡Increíble! Comparte tu pantalla, ¡comencemos!	<p><i>El alumno comparte su pantalla.</i></p> <p><i>Abre la actividad del alumno, hace un remix al código y le agrega más código.</i></p>
<p>La maestra finaliza la presentación de diapositivas</p> 	
La maestra deja de compartir pantalla	
ACTIVIDAD GUIADA POR EL ALUMNO - 8 minutos	
<ul style="list-style-type: none"> • Pídele al alumno que presione la tecla ESC para volver al panel • Guía al alumno para que comience a compartir la pantalla. • La maestra cambia a pantalla completa 	
<p><u>ACTIVIDAD</u></p> <ul style="list-style-type: none"> • El alumno programa para crear el juego completo. • Agrega retos opcionales para el alumno. 	
Acción de la maestra	Acción del alumno

Paso 3:

Actividad guiada por el alumno

¿Recuerdas cómo se crean los sprites?

Usemos la función **createSprite()** para crear las paredes en el juego.

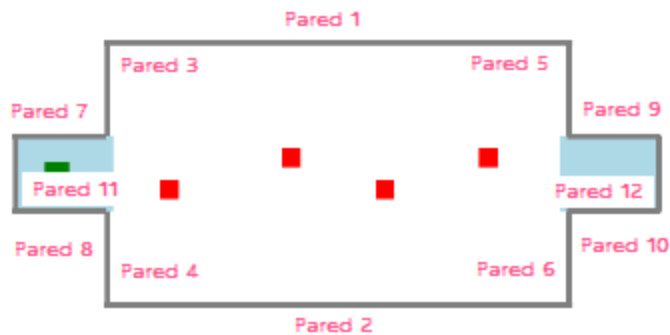
Ayuda al alumno a crear las paredes del túnel usando sprites.

El alumno abre [Actividad del alumno 2](#)

REA

Usando la función createSprite().

El alumno escribe el código para crear y colocar las paredes del túnel en el juego.



```

1  var wall1 = createSprite(190,120,250,3);
2  var wall2 = createSprite(190,260,250,3);
3  var wall3 = createSprite(67,145,3,50);
4  var wall4 = createSprite(67,235,3,50);
5  var wall5 = createSprite(313,145,3,50);
6  var wall6 = createSprite(313,235,3,50);
7  var wall7 = createSprite(41,170,50,3);
8  var wall8 = createSprite(41,210,50,3);
9  var wall9 = createSprite(337,210,50,3);
10 var wall10 = createSprite(337,170,50,3);
11 var wall11 = createSprite(18,190,3,40);
12 var wall12 = createSprite(361,190,3,40);
13

```

Ayuda al alumno a crear el recuadro verde, que se puede controlar con las teclas de flecha derecha e izquierda.

Vuelve a mencionar la programación condicional y los eventos clave para el alumno.

El alumno escribe el código para crear un sprite de recuadro rojo y lo controla, usando las teclas de flecha derecha e izquierda..



Crear el recuadro verde

```
14 var ding = createSprite(40,190,13,13);
15 ding.shapeColor = "green";
16
```

Controlar el recuadro verde

```
54 if(keyDown("right")){
55     ding.x = ding.x + 2;
56 }
57 if(keyDown("left")){
58     ding.x = ding.x - 2;
59 }
60
```

Lo estás haciendo genial hasta ahora.
Haz que el alumno cree los cuadros
rojos que rebotan de las paredes
superior e inferior.

*Los alumnos crean cuatro
cuadros verdes que rebotan
de las paredes superior e
inferior.*

```

5
6  var dong1 = createSprite(100,130,10,10);
7  dong1.shapeColor = "red";
8  var dong2 = createSprite(215,130,10,10);
9  dong2.shapeColor = "red";
10 var dong3 = createSprite(165,250,10,10);
11 dong3.shapeColor = "red";
12 var dong4 = createSprite(270,250,10,10);
13 dong4.shapeColor = "red";
14
15
16  dong1.velocityY = 8;
17  dong2.velocityY = 8;
18  dong3.velocityY = -8;
19  dong4.velocityY = -8;
20

```

```

43
44  dong1.bounceOff(wall1);
45  dong1.bounceOff(wall2);
46  dong2.bounceOff(wall1);
47  dong2.bounceOff(wall2);
48  dong3.bounceOff(wall1);
49  dong3.bounceOff(wall2);
50  dong4.bounceOff(wall1);
51  dong4.bounceOff(wall2);
52

```

Ayuda al alumno a agregar lógica de modo que el recuadro rojo se restablezca en su posición, cuando toque cualquiera de los recuadros verdes.

El alumno escribe el código para restablecer la posición de la pelota roja cuando toca cualquiera de los recuadros verdes.

```

60
61  if(ding.isTouching(wall11)||
62     ding.isTouching(wall12)||
63     ding.isTouching(dong1)||
64     ding.isTouching(dong2)||
65     ding.isTouching(dong3)||
66     ding.isTouching(dong4))
67  {
68     ding.x = 40;
69     ding.y = 190;
70     count = count + 1;
71  }
72

```

Guía al alumno a crear un sistema de puntuación donde se contabilicen las muertes.

El alumno crea una variable de conteo y la aumenta después de cada muerte.

El alumno lo muestra como muertes.

```
30
31  var count = 0;
32
```

```
61  if(ding.isTouching(wall11)||
62      ding.isTouching(wall12)||
63      ding.isTouching(dong1)||
64      ding.isTouching(dong2)||
65      ding.isTouching(dong3)||
66      ding.isTouching(dong4))
67  {
68      ding.x = 40;
69      ding.v = 190;
70      count = count + 1;
71  }
```

Ayuda al alumno a elegir un sonido, cargarlo y luego reproducirlo en un bucle.

El alumno usa recursos de la [Actividad del alumno 2](#)

El alumno elige un sonido, lo carga y luego lo reproduce en un bucle.

```
32
33  playSound("sound123.mp3", true);
34
```

(Opcional) Puedes agregar niveles a tu juego usando gameState (estado del juego).

Aumenta la velocidad de los recuadros móviles en niveles más avanzados.

El alumno escribe el código para crear niveles más avanzados en el juego.

Antes de que terminemos la clase, vamos a descargar un programa llamado Visual Studio Code.

A partir de la siguiente clase, programaremos usando este programa.

El alumno abrirá la [Actividad del alumno 4](#) y comenzará a descargar VS code de acuerdo a su sistema operativo

The screenshot shows the Visual Studio Code download page. It features three main sections for different operating systems: Windows, Linux, and Mac. Each section has a download icon and a list of available installer formats and their bitness. For Windows, it lists 'User Installer' and 'System Installer' in both 64 bit and 32 bit formats, along with a '.zip' file. For Linux, it lists '.deb' (Debian, Ubuntu), '.rpm' (Red Hat, Fedora, SUSE), and '.tar.gz' in 64 bit format, plus a link to the 'Snap Store'. For Mac, it lists 'Mac' in 64 bit format for macOS 10.10+.



Solo descargaremos el programa, la instalación se hará en la siguiente clase.

Guía al alumno para que deje de compartir la pantalla

Hora del quiz – haz clic en el quiz de clase

Pregunta	Respuesta
<p>Para hacer que los cuadros rojos (dong) se muevan alternativamente de forma vertical, debemos:</p> <ul style="list-style-type: none"> A. Debemos darle a cada uno valores de velocityX opuestos B. Debemos darle a cada uno valores velocityY opuestos C. Darle a cada uno valores opuestos de x 	<p>B.</p>

D. Darle a cada uno valores opuestos de y	
<p>Identifica la sintaxis correcta si queremos escuchar música mientras jugamos el juego:</p> <p>A. playSound("sound.mp3", true); después de la función draw()</p> <p>B. playSound("sound.mp3", false); después de la función draw()</p> <p>C. playSound("sound.mp3", true); antes de la función draw()</p> <p>D. playSound("sound.mp3", false); antes de la función draw()</p>	C.
<p>Identifica la sintaxis correcta para mover el ding a la derecha al presionar la tecla derecha.</p> <p>A. if(keyDown("right")){ ding.x = ding.x + 2; }</p> <p>B. if(keyDown("right")){ ding.x = ding.y +2; }</p> <p>C. if(keyDown("right")){ ding.y = ding.y + 2; }</p> <p>D. if(keyDown("right")){ ding.x = ding.y - 2; }</p>	A.
Finaliza el panel del quiz	
CONCLUSIÓN - 5 minutos	

<div>  </div> <p>La maestra inicia la presentación de diapositivas desde la diapositiva 18 a la 28</p>	
Acción de la maestra	Acción del alumno
<p>Ejecuta la presentación de la diapositiva 18 a la 28.</p> <p>Los siguientes son los resultados de la sesión de Rompiendo el hielo:</p> <ul style="list-style-type: none"> • Explica los hechos y trivia • Reto de la siguiente clase • Proyecto del día • Actividad adicional 	<p>Guía al alumno para que desarrolle el proyecto y lo comparta con nosotros.</p>
<div>  </div> <p>La maestra finaliza la presentación de diapositivas</p>	
<p style="text-align: center;"><u>RETROALIMENTACIÓN</u></p> <ul style="list-style-type: none"> • Felicita al alumno por su esfuerzo. • Identifica 2 fortalezas y 1 área de oportunidad para el alumno. 	
Acción de la maestra	Acción del alumno
<p>Paso 4: Conclusión</p> <p>Entonces, en esta clase revisamos los conceptos que hemos aprendido hasta ahora.</p> <p>¿Cómo fue tu experiencia?</p> <p>Ahora puedes agregar niveles más altos de retos al juego e invitar a tus amigos a jugarlos.</p> <p>¡Felicidades, ahora eres un desarrollador de juegos!</p>	<p>REA: Variada</p>
	-

<p>¡Felicidades! Has alcanzado una meta. ¿Estás listo para enfrentar uno de los retos más difíciles hasta ahora?</p> <p>Para completar con éxito este reto, tienes que aplicar los constructos de programación aprendidos durante las clases anteriores y crear “el juego más difícil del mundo”.</p>	
<p>Visión del proyecto EL JUEGO MÁS DIFÍCIL DEL MUNDO</p> <p>Objetivo del proyecto: Hoy revisaste los conceptos de las clases anteriores y creaste el juego más difícil del mundo.</p> <p>En este proyecto diseñarás tu propio juego.</p> <p>Historia: Cada juego tiene unos cuantos componentes simples...</p> <p>El jugador debe tener un objetivo, debe darte obstáculos que superar y debe ofrecer comentarios en cómo vas durante el juego. Además, los buenos juegos siempre tienen una historia de fondo.</p> <p>Con todo esto para considerar, diseña tu propio juego con tus propios personajes, objetivos, obstáculos, comentarios e historia.</p> <p>Estoy muy emocionada de ver la solución de tu proyecto y sé que lo harás muy bien.</p> <p>¡Adiós!</p>	<p><i>Los alumnos abordan el proyecto con la maestra.</i></p> <p>Esperamos que el alumno aplique su creatividad en el proyecto.</p>
<p style="text-align: center;">La maestra hace clic en ✕ Finalizar Clase</p>	
<p>ACTIVIDADES ADICIONALES</p>	

- Pide al alumno que haga clic en ESC para regresar al panel.
- Guía al alumno para que comparta su pantalla.
- La maestra cambia a pantalla completa.

Detalles de la actividad		Solución/Lineamientos
Actividad adicional	<p><i>Alienta al alumno a escribir lo que aprendió en su diario de reflexión, usando markdown.</i></p> <p>Usa las siguientes preguntas como guía:</p> <ul style="list-style-type: none"> • ¿Qué pasó hoy? <ul style="list-style-type: none"> - Describe qué pasó - El código que escribí • ¿Cómo me sentí después de la clase? • ¿Qué he aprendido acerca de programación y desarrollo de juegos? • ¿Qué aspectos de la clase me ayudaron? • ¿Qué me pareció difícil? 	<p><i>El alumno usa markdown para escribir sus reflexiones.</i></p>

Actividad	Nombre de la actividad	Enlaces
Actividad del alumno 1	Juego	https://studio.code.org/projects/gamelab/GnUYFxJSdK0_0i2HcQj7hc64tiPM8_wFsguRdKEPp_Q
Actividad del alumno 2	Actividad de proyecto de clase	https://studio.code.org/projects/gamelab/kMc7OERGyJWZVWDkC5V918EBLOikdMZuVXERnZmBIW0
Actividad del alumno 3	Recurso de sonido	https://freesound.org/people/djgriffin/sounds/251284/

Actividad del alumno 4	Enlace de VS code	https://code.visualstudio.com/download
Actividad de la maestra 1	Juego	https://studio.code.org/projects/gamelab/JM1Qr6m1M4flrd07bbg_9hOwXHWRZq3UyPsoUsWMj08
Actividad de la maestra 2	Enlace de referencia	https://studio.code.org/projects/gamelab/vVRuW8jBwojYV4rTU8v9YAOAnJYBlizrKvU6xt1iYsw
Actividad de la maestra 3	Plantilla del código	https://studio.code.org/projects/gamelab/vVRuW8jBwojYV4rTU8v9YCYZt-8nhxeCTiLIQjIMdh2s
Actividad de la maestra 4	Código de referencia	https://studio.code.org/projects/gamelab/dyiP-762JO6IfYxdHfeDBAnTA4Ry19InhWIHUmACcl0
Enlace de referencia de apoyo visual de la maestra	Enlace de referencia visual con notas	https://s3-whjr-curriculum-uploads.whjr.online/4378fcc3-e79a-4eaf-9914-3b48001de0bd.html
Enlace de referencia de ayuda visual de la maestra	Enlace de referencia visual sin notas	https://s3-whjr-curriculum-uploads.whjr.online/4f3d5097-5587-44f5-8c8e-11c3763bc4a8.html
Quiz enclase-Referencia de la maestra	Quiz en clase	https://s3-whjr-curriculum-uploads.whjr.online60df172d-200b-4c29-857e-e1594a17b1bc.pdf