Adrián Jiménez Villarreal

71904816P | uo227602

Videojuegos-Infinite

Ampliaciones

Contenido

[Ampliaciones 2](#_Toc468658677)

[Varias vidas 2](#_Toc468658678)

[Modificación del Jugador 2](#_Toc468658679)

[Modificación de los controles 2](#_Toc468658680)

[Modificación de la escena de juego 2](#_Toc468658681)

[Scroll en el eje Y 2](#_Toc468658682)

[Modificación de la escena de juego 3](#_Toc468658683)

[Capacidad de disparo 3](#_Toc468658684)

[Nuevo tipo de enemigo 3](#_Toc468658685)

[Turbo 3](#_Toc468658686)

[Pinchos 4](#_Toc468658687)

[Niveles 4](#_Toc468658688)

[Control el modo plataformas 4](#_Toc468658689)

# Ampliaciones

## Varias vidas

Para realizar esta ampliación primero deberemos modificar el objeto Jugador, posteriormente la capa de controles y finalmente la escena del juego.

### Modificación del Jugador

Añadir una variable vidas inicializada a 3 y la siguiente función.

|  |
| --- |
| }, restaVida: **function**(){  **this**.vidas--; } |

### Modificación de los controles

Habrá que crear una etiqueta para las vidas, e inicializarla de la siguiente manera.

|  |
| --- |
| *// Contador Vidas* **this**.etiquetaVidas = **new** cc.LabelTTF(**"Vidas: 3"**, **"Helvetica"**, 20); **this**.etiquetaVidas.setPosition(cc.p(size.width - size.width+90, size.height - 20)); **this**.etiquetaVidas.fillStyle = **new** cc.Color(0, 0, 0, 0); **this**.addChild(**this**.etiquetaVidas); |

También habrá que crear un método para actualizar esta etiqueta cada vez que pierda una vida el jugador.

|  |
| --- |
| },actualizarVidas:**function**(vidas){  **this**.etiquetaVidas.setString(**"Vidas: "** + vidas); } |

### Modificación de la escena de juego

Habrá que modificar la colisión del jugador con enemigos, para que pierda vidas cuando chocas con ellos.

|  |
| --- |
| ,collisionJugadorConEnemigo:**function** (arbiter, space) {  **var** capaControles = **this**.getParent().getChildByTag(idCapaControles);  **this**.jugador.restaVida();  **var** shapes = arbiter.getShapes();  **this**.formasEliminar.push(shapes[1]);  **if**(**this**.jugador.vidas<=0){  cc.director.pause();  cc.director.runScene(**new** GameOverLayer());  }  capaControles.actualizarVidas(**this**.jugador.vidas); } |

## Scroll en el eje Y

Para realizar esta ampliación simplemente deberemos modificar como se mueve la cámara, no solo en el eje x sino también en él y.

### Modificación de la escena de juego

En el método update se controlará la cámara de la siguiente manera, así seguirá al jugador tanto en el eje horizontal como el vertical.

|  |
| --- |
| *// Ampliacion Scroll eje Y* **var** posicionXCamara = **this**.jugador.body.p.x - **this**.getContentSize().width/2; **var** posicionYCamara = **this**.jugador.body.p.y - **this**.getContentSize().height/2;  **if** ( posicionXCamara < 0 ){  posicionXCamara = 0; } **if** ( posicionXCamara > **this**.mapaAncho - **this**.getContentSize().width ){  posicionXCamara = **this**.mapaAncho - **this**.getContentSize().width; }  **if** ( posicionYCamara < 0 ){  posicionYCamara = 0; } **if** ( posicionYCamara > **this**.mapaAlto - **this**.getContentSize().height ){  posicionYCamara = **this**.mapaAlto - **this**.getContentSize().height ; }  **this**.setPosition(cc.p( - posicionXCamara , - posicionYCamara)); |

## Capacidad de disparo

## Nuevo tipo de enemigo

## Turbo

Para realizar esta ampliación se necesita insertar un botón de turbo en la capa de controles, controlar la pulsación sobre este y una función en el jugador que le dé un impulso.

### Inclusión del botón

|  |
| --- |
| *// BotonTurbo* **this**.spriteBotonTurbo = cc.Sprite.create(res.boton\_turbo\_png); **this**.spriteBotonTurbo.setPosition(  cc.p(size.width\*0.8, size.height\*0.8));  **this**.addChild(**this**.spriteBotonTurbo); |

### Control de la pulsación

|  |
| --- |
| procesarMouseDown:**function**(event) {  **var** instancia = event.getCurrentTarget();  **var** areaBoton = instancia.spriteBotonSaltar.getBoundingBox();  **var** areaTurbo = instancia.spriteBotonTurbo.getBoundingBox();   *// La pulsación cae dentro del botón* **if** (cc.rectContainsPoint(areaBoton,  cc.p(event.getLocationX(), event.getLocationY()) )){   *// Accedemos al padre (Scene), pedimos la capa con la idCapaJuego* **var** gameLayer = instancia.getParent().getChildByTag(idCapaJuego);  *// tenemos el objeto GameLayer* gameLayer.jugador.saltar();  }  **if** (cc.rectContainsPoint(areaTurbo,  cc.p(event.getLocationX(), event.getLocationY()) )){  **var** gameLayer = instancia.getParent().getChildByTag(idCapaJuego);  gameLayer.jugador.turbo();  } |

### Inclusión de la función para impulsar

Se debe crear primero una variable con los usos máximos del turbo.

|  |
| --- |
| },turbo: **function**(){  **if**(**this**.usosTurbo>0){  **this**.body.applyImpulse(cp.v(5000, 0), cp.v(0, 0));  **this**.usosTurbo--;  } } |

## Pinchos

## Niveles

## Control el modo plataformas