

Shoot Operation  
Desarrollo de sistemas hipermedia  
Universidad de Cádiz

May 17, 2021

Alumnos: Alejandro Serrano Fernández  
Pedro Antonio Navas Luque  
Profesor: José Miguel Mota Macías

# Contents

<b>1</b>	<b>Introducción</b>	<b>3</b>
<b>2</b>	<b>Idea del proyecto</b>	<b>3</b>
<b>3</b>	<b>Instrucciones de uso</b>	<b>3</b>
<b>4</b>	<b>Herramientas Empleadas</b>	<b>4</b>
4.1	Photoshop . . . . .	4
4.2	Vuforia . . . . .	4
4.3	Blender . . . . .	4
4.4	Visual Studio Code . . . . .	5
4.5	Unity . . . . .	5
<b>5</b>	<b>Distribución del trabajo</b>	<b>5</b>
<b>6</b>	<b>Prefabs</b>	<b>6</b>
6.1	Dragón . . . . .	6
6.2	Castillo . . . . .	6
6.3	Robot Asesino . . . . .	7
6.4	Monstruo Cerdo . . . . .	7
6.5	Bastón Mágico . . . . .	8
<b>7</b>	<b>Scripts</b>	<b>8</b>
<b>8</b>	<b>Interfaz in Game</b>	<b>9</b>
<b>9</b>	<b>Dificultades durante el desarrollo</b>	<b>10</b>
<b>10</b>	<b>Posibles Mejoras</b>	<b>10</b>
<b>11</b>	<b>Componentes Adicionales</b>	<b>11</b>
<b>12</b>	<b>Paquetes Adicionales</b>	<b>12</b>

# 1 Introducción

En este documento se describe todos los procesos e ideas implicadas en el desarrollo del proyecto final de la asignatura "Desarrollo de sistemas hipermedia". La idea general de este proyecto es aplicar la Realidad Aumentada en el desarrollo de los videojuegos, para que el público o clientes, puedan obtener mayor experiencia e interacción en éstos.

# 2 Idea del proyecto

Nuestra idea es desarrollar un videojuego, aplicando la realidad aumentada, donde el usuario colocará targets (Tarjetas) por toda una zona (Ej.Casa, Edificio...). Estos targets darán lugar a distintos monstruos que el usuario deberá eliminar con el arma recibida en el inicio del juego. El juego acabará cuando el usuario haya derrotado a todos los enemigos y haya derrumbado la torre final o cuando su vida baje a cero.

Solo puede jugarse por una persona, aunque está permitido que varios usuarios puedan competir entre ellos para ver quién consigue una mejor puntuación (menor tiempo).

# 3 Instrucciones de uso

- a. Descargar e imprimir los targets que se ponen a disposición para el usuario. El tamaño ideal será el estándar A4.
- b. Descargar e instalar la apk ofrecida de Shoot Operation.
- c. Colocar o esconder los targets por toda la zona de juego. Estos pueden colocarse o esconderse por el mismo usuario o por otro.
- d. Iniciar la aplicación y pulsar en iniciar juego.
- e. El usuario, nada más iniciar, obtendrá un arma con el que podrá eliminar a los enemigos, aunque no la torre final. Para eliminar la torre, es necesario que el usuario encuentre el target que le permite obtener el bastón mágico. Este bastón tiene la energía y el daño suficiente para acabar con ella, aunque cerca aguarda un enorme dragón que no permitirá que acabes tan rápido con ella.
- f. Pulsando el botón de pausa se desplegaran tres opciones: Continuar, volver al menu principal y volver a empezar la partida.
- g. El tiempo se podrá observar en todo momento durante la partida.
- h. Si varios usuarios están jugándolo, ganará aquél que menor tiempo haya empleado para completar el juego.

## 4 Herramientas Empleadas

### 4.1 Photoshop

Adobe Photoshop es un editor de fotografías desarrollado por Adobe Systems Incorporated. Usado principalmente para el retoque de fotografías y gráficos. Esta herramienta nos ha permitido generar las imágenes del inicio del juego, su icono y botones.

### 4.2 Vuforia

Vuforia es un kit de desarrollo de software de realidad aumentada para dispositivos móviles que permite la creación de aplicaciones de realidad aumentada. Utiliza tecnología de visión por computadora para reconocer y rastrear imágenes planas y objetos 3D en tiempo real. Esta herramienta nos permite que los targets puedan ser reconocidos por el ordenador o móvil.

### 4.3 Blender

Blender es un programa informático multiplataforma, dedicado especialmente al modelado, iluminación, renderizado, la animación y creación de gráficos tridimensionales. También de composición digital utilizando la técnica procesal de nodos, edición de vídeo, escultura y pintura digital. Esta herramienta nos ha permitido diseñar y modelar el bastón utilizado en el juego para destruir la torre final.

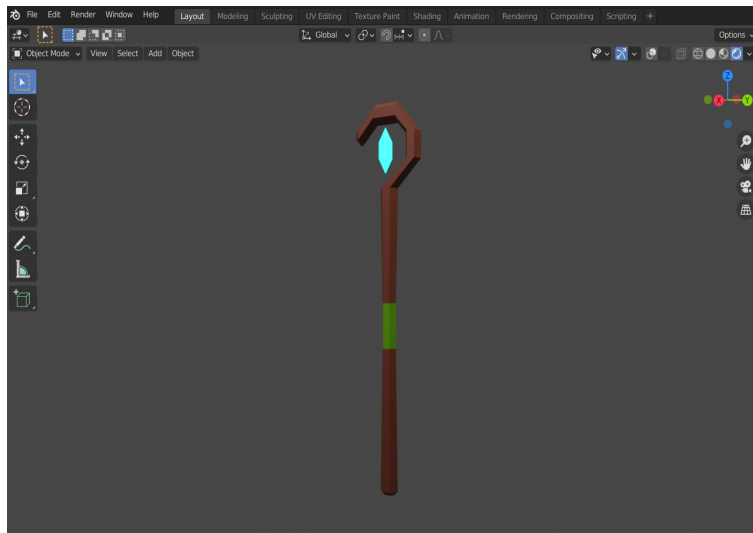


Figure 1: Bastón diseñándose en blender

## 4.4 Visual Studio Code

Visual Studio Code es un editor de código fuente desarrollado por Microsoft. Esta herramienta nos permite escribir código en C sharp. Gracias a su auto completado, podemos escribir código en menor tiempo.

## 4.5 Unity

Unity es un motor de videojuego multiplataforma creado por Unity Technologies. Esta herramienta nos ha permitido desarrollar gran parte de nuestro juego (Ej.Colocar enemigos, programación de disparos...)

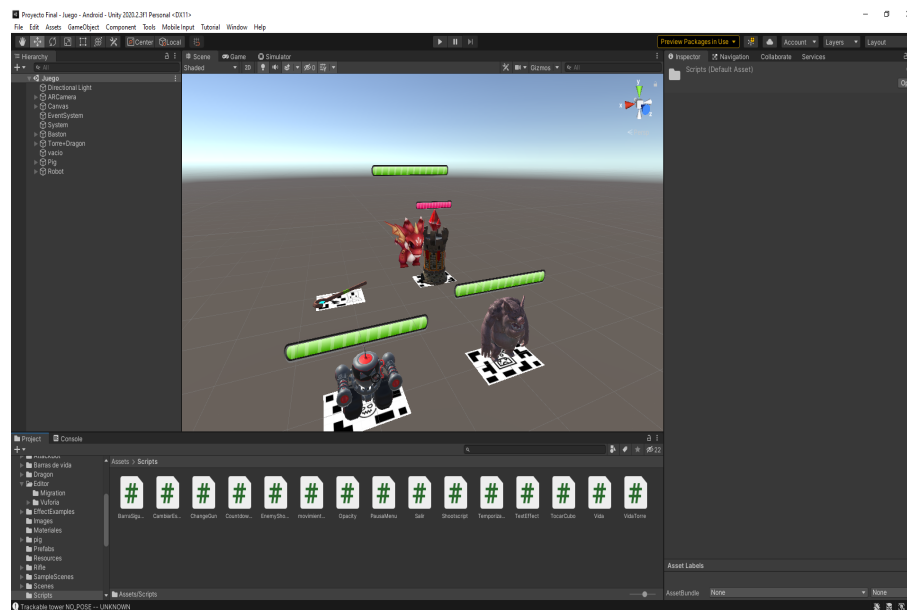


Figure 2: Proyecto desarrollándose en Unity

## 5 Distribución del trabajo

**Alejandro** Encargado de la programación de algunos scripts del juego, diseñar interfaz y estilo del juego (Diseñar imágenes en Photoshop, interfaz del canvas...).

**Pedro** Encargado de la programación de la física del juego (Scripts de disparos, animaciones, efectos de partículas...) y del diseño en blender.

La documentación ha sido desarrollada entre ambos.

## 6 Prefabs

### 6.1 Dragón

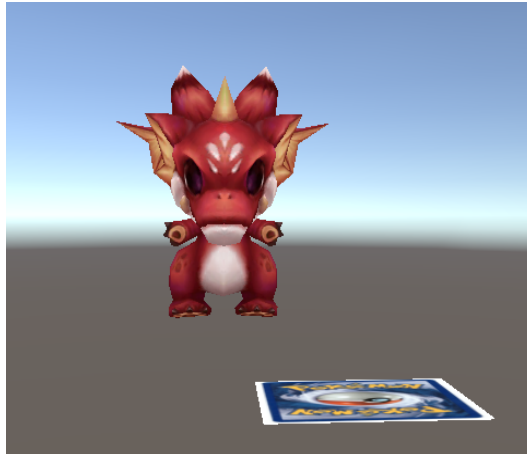


Figure 3: Prefab del dragón

### 6.2 Castillo

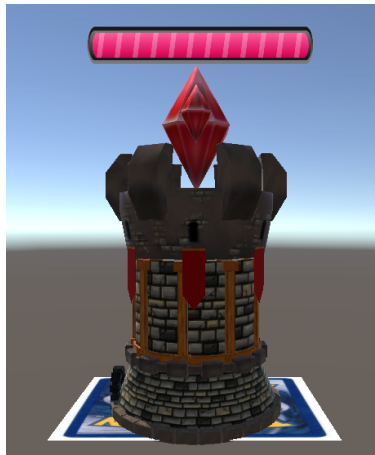


Figure 4: Prefab del castillo

### 6.3 Robot Asesino



Figure 5: Prefab del robot asesino

### 6.4 Monstruo Cerdo



Figure 6: Prefab del monstruo cerdo

## 6.5 Bastón Mágico

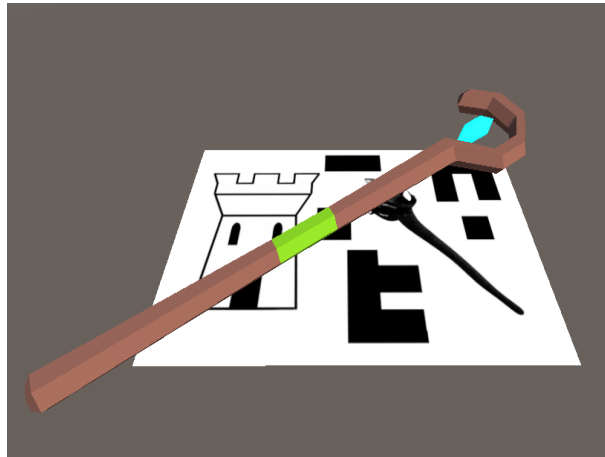


Figure 7: Prefab del bastón mágico

## 7 Scripts

Breve descripción de los scripts que usamos en el proyecto.

**BarraSigueCamara.cs** Rota la barra de vida de los enemigos para que siempre esté siguiendo a la cámara.

**CambiarEscenario.cs** Como su nombre indica, contiene las funciones que permiten el cambio de escenas.

**ChangeGun.cs** Intercambia nuestra arma actual cada vez que pulsamos el botón de intercambio.

**CountdownTimer.cs** Clase que ofrece distintas funcionalidades para trabajar con el tiempo.

**EnemyShoot.cs** Este script se encarga de crear los proyectiles disparados por los diferentes enemigos y el intervalo entre cada proyectil.

**movimientobala.cs** Aporta el movimiento a las balas y detecta la colisión con el jugador.

**Opacity.cs** Se encarga de dar opacidad a una imagen.

**PausaMenu.cs** Nos visibiliza el menú de pausa.

**Salir.cs** Salir del juego cuando se presiona el botón Salir.



**Shootscript.cs** Este script está ligado al botón de disparo y se encarga de detectar si le hemos dado o no al enemigo, si es así entonces lo reflejara al usuario con un efecto sobre el enemigo y llamara al script que se encarga de reducirles la vida. Desactiva el botón de disparo durante un momento para que el jugador no pueda disparar todo el rato. Además se encarga de recoger el bastón mágico cuando lo encontremos y habilitar la posibilidad de cambiar nuestro arma. Y dependiendo del arma actual del jugador le aporta un efecto de partículas diferente al disparo.

**Temporizador.cs** Cuenta el tiempo total que el jugador lleva jugando.

**TextEffect.cs** Efecto del texto en la pantalla inicial del juego. Baja y sube la opacidad del texto.

**Vida.cs** Maneja la vida de los enemigos y la del jugador. Cuando disparamos a un enemigo, éste se encarga de rebajársela y destruirlo cuando llega a cero su vida. Actúa de la misma manera para el jugador cuando es golpeado por los proyectiles de los enemigos.

**VidaTorre.cs** Maneja la vida de la torre. Cuando disparamos a la torre, éste se encarga de rebajársela.

## 8 Interfaz in Game

Lo que vemos durante el juego, una vez ya comenzado es:

- a. En la parte superior izquierda: nuestra barra de vida.
- b. En la parte superior centro: el tiempo.
- c. En la parte superior derecha tenemos el botón de pausa.
- d. En el centro de la pantalla tenemos la mira con la cual podemos saber donde estamos apuntando.
- e. En la parte inferior derecha tenemos tanto nuestra arma actual que estamos usando, como el botón de disparar.
- f. Una vez obtenido el bastón mágico aparecerá encima del botón de disparo un botón para cambiar nuestra arma actual por la otra. Además cuando encontremos a un enemigo nos saldrá su barra de vida encima del mismo.

Aquí algunas capturas del juego:

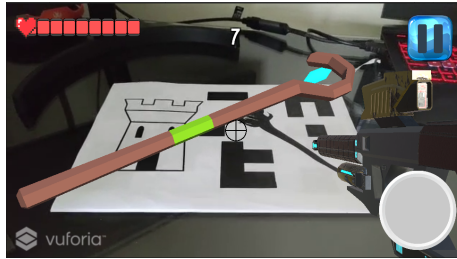


Figure 8: Interfaz con el Target del bastón

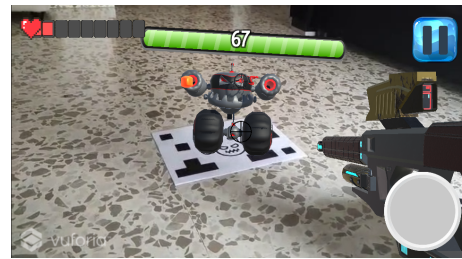


Figure 9: Interfaz cuando perdemos vida

## 9 Dificultades durante el desarrollo

Durante el desarrollo del juego hemos enfrentado diversos problemas con los que hemos tenido que tratar.

Algunos de ellos son:

- a. Uno de los primeros problemas a los que nos enfrentamos fue a la hora de exportar el bastón mágico de blender a unity ya que uno de los materiales no aparecía al exportarlo, en concreto el material de la gema, el cual era Emission. Después de intentar exportarlo de diversas maneras descubrimos que era problema del propio tipo (Surface: Emission) por tanto investigando descubrimos que podíamos conseguir el mismo efecto modificando el material por defecto (Principled BSDF).
- b. Otro problema que encontramos fue controlar cuando un enemigo esta trakeado o no para que lance proyectiles, esto lo solventamos investigando un poco sobre Vuforia y haciendo que cuando la imagen es trakeada llamamos a una función que pone una variable booleana a true, y cuando deja de trakearlo la pone a false.

## 10 Posibles Mejoras

- a. Utilizar otros kits de desarrollo de realidad aumentada, como por ejemplo AR Foundation, que ofrece una mejor optimización
- b. Añadir más enemigos. Solo hemos implementado 3. La idea es implementar más enemigos, cada uno con ataques distintos.
- c. Mejorar la interfaz del usuario
- d. Añadir nuevas funcionalidades al juego. Por ejemplo, más armas, camuflaje de armas...
- e. Desarrollo multiplataforma. Actualmente nuestro proyecto está únicamente disponible para Android.

- f. Movimiento de los enemigos.
- g. Implementar munición.
- h. Mejor sistema de vida y ataque. Buscar aquella que más equilibre el juego y ofrezca una mejor experiencia.
- i. Añadir efectos de sonido.

## 11 Componentes Adicionales

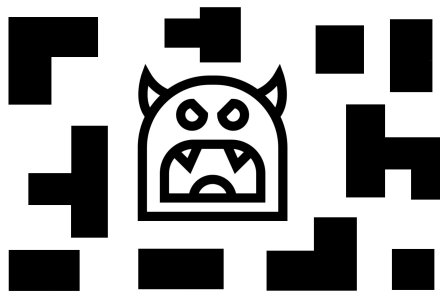


Figure 10: Target del monstruo cerdo

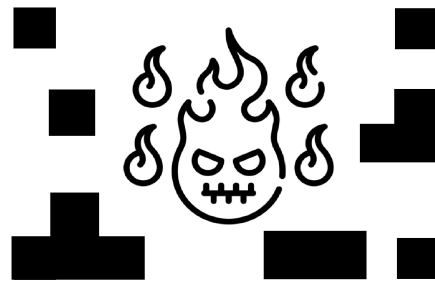


Figure 11: Target del robot

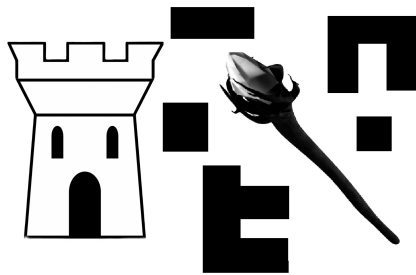


Figure 12: Target del bastón

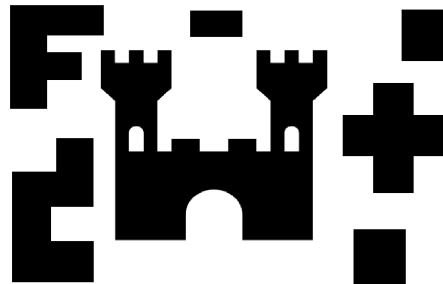


Figure 13: Target Dragón y Torre

## 12 Paquetes Adicionales

Para este proyecto hemos añadido varios paquetes de la Asset Store, entre ellos varios con sistemas de partículas. Muchas de las que hemos utilizado o bien las hemos creado nosotros o modificado a partir de estas de los paquetes. Además los modelos de monstruos, la torre y el fusil los hemos sacado también de la Asset Store.

Links de los paquetes y assets empleados:

**Robot:** <https://assetstore.unity.com/packages/3d/characters/robots/attack-bot-15120>

**Cerdo:** <https://assetstore.unity.com/packages/3d/characters/creatures/pbr-orc-pig-109248>

**Dragón:** <https://assetstore.unity.com/packages/3d/characters/small-red-dragon-52959>

**Torre:** <https://assetstore.unity.com/packages/3d/environments/fantasy/awesome-stylized-mage-tower-53793>

**Sci-Fi Rifle Low Poly:** <https://assetstore.unity.com/packages/3d/props/guns/sci-fi-rifle-low-poly-99671>

**Unity Particle Pack:** <https://assetstore.unity.com/packages/essentials/tutorial-projects/unity-particle-pack-127325>

**Standard Assets:** <https://assetstore.unity.com/packages/essentials/asset-packs/standard-assets-for-unity-2018-4-32351>