

**MP_0489. Programación
multimedia y dispositivos móviles**
UF4. Análisis de motores de juegos

4.2. Trabajar con Unity

Índice

☰	Objetivos	3
☰	Instalación	4
☰	Área de trabajo	10
☰	Herramientas	17
☰	Resumen	23

Objetivos

Con esta unidad perseguimos los siguientes objetivos:

1

Aprender a instalar Unity, un motor para el desarrollo de videojuegos dentro del entorno Android.

2

Descubrir cómo funciona la interfaz de usuario de Unity.

¡Ánimo y adelante!

Instalación

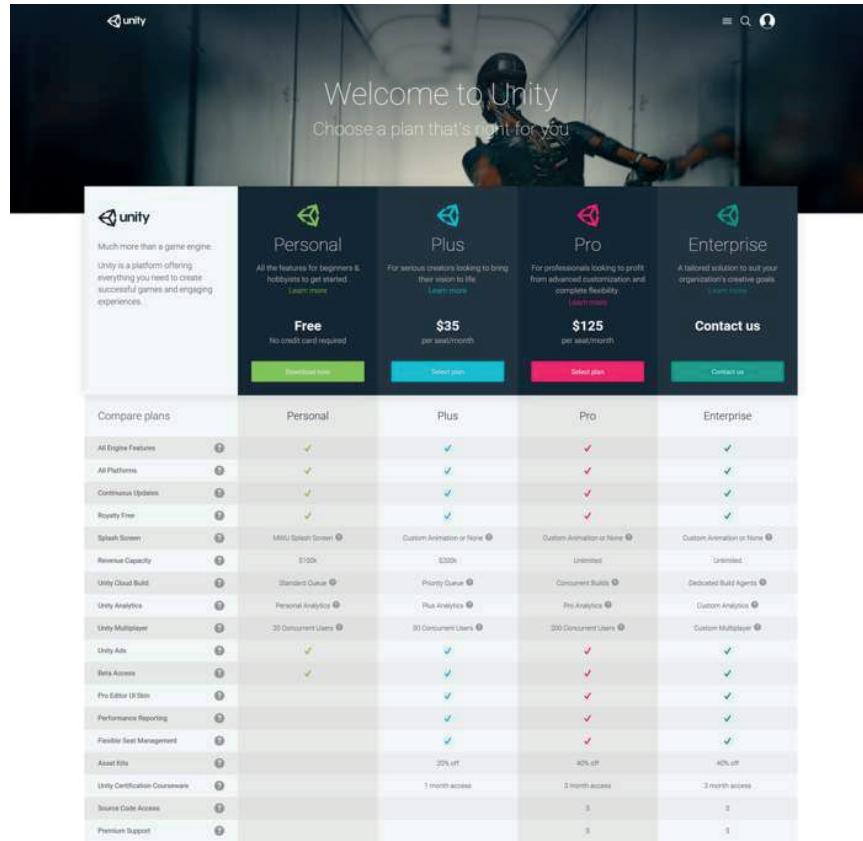
Lo primero que debemos aprender para trabajar con Unity es, lógicamente, **cómo instalarlo**.

Para ello, comenzaremos por **ir a la página de Unity y descargarnos el programa**.

 Aquí podemos descargarlo: <https://unity3d.com/get-unity>

1

En la página de Unity encontraremos cuatro opciones. Para aprender y dar nuestros primeros pasos, elegimos **instalar la versión Free**.



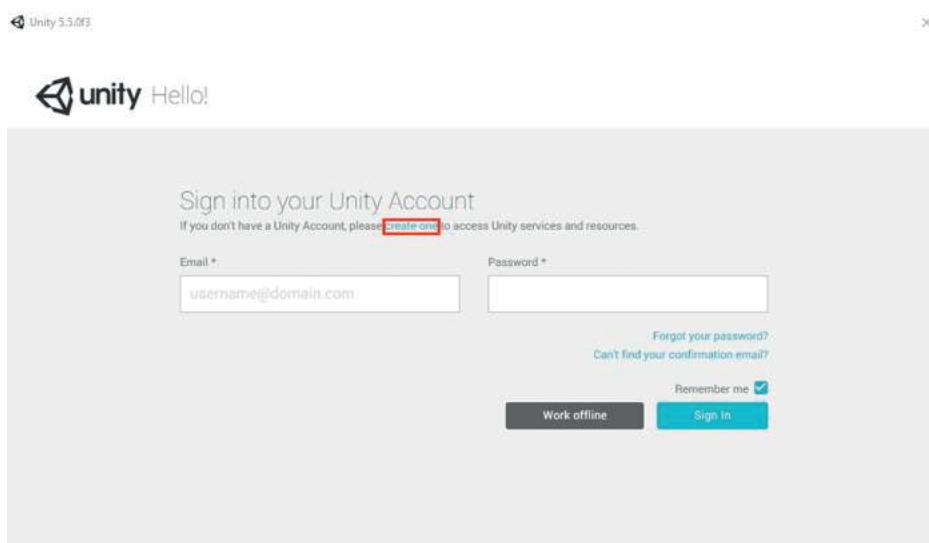
2

Descargamos la unidad y la instalamos, siguiendo las instrucciones que vemos en pantalla.



3

Esperamos a que finalice la instalación y luego **iniciamos Unity**; se nos pedirá que iniciemos sesión antes de continuar.



4

Necesitaremos crear una cuenta gratuita de Unity para poder usar el software. La creamos rellenando el siguiente formulario:

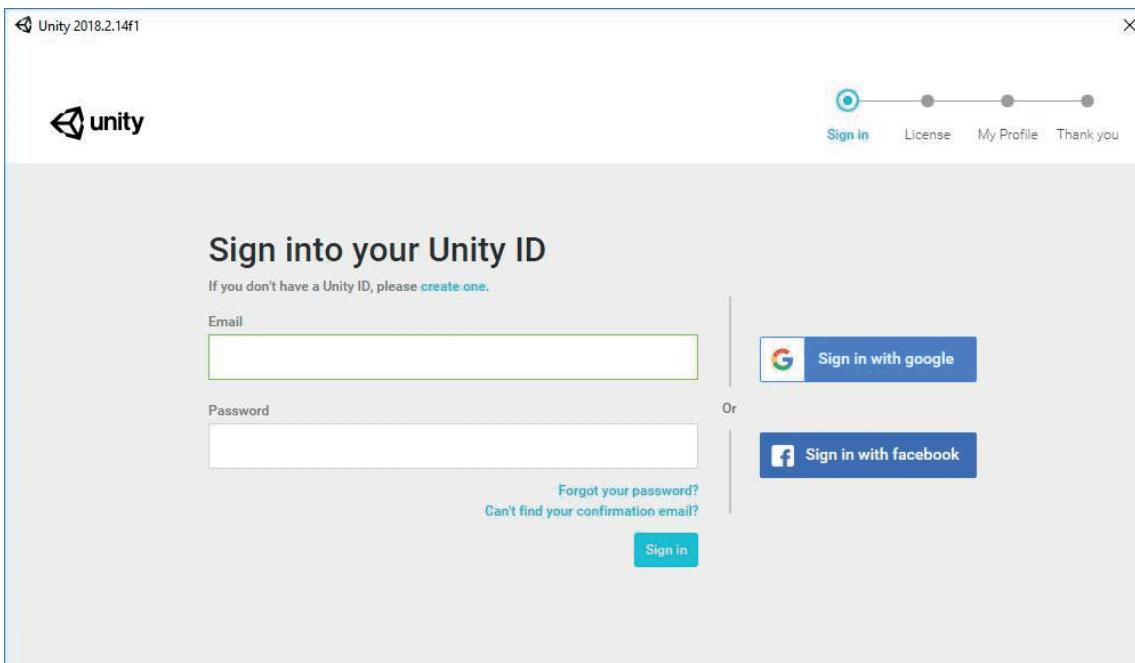
The screenshot shows the 'Create a Unity ID' page. It includes fields for Email, Password, Username, Full Name, and Country. Below these are two checkboxes: 'I agree to the Unity Terms of Use and Privacy Policy' and 'Get Unity news, discounts and more!'. At the bottom are 'Create a Unity ID' and 'Already have a Unity ID?' buttons.

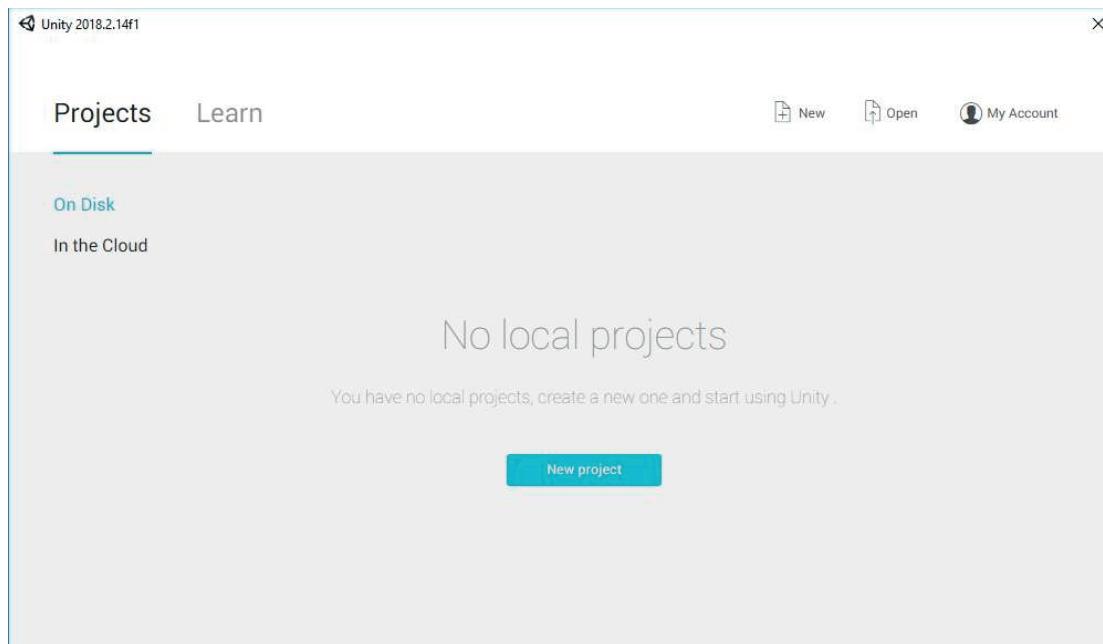
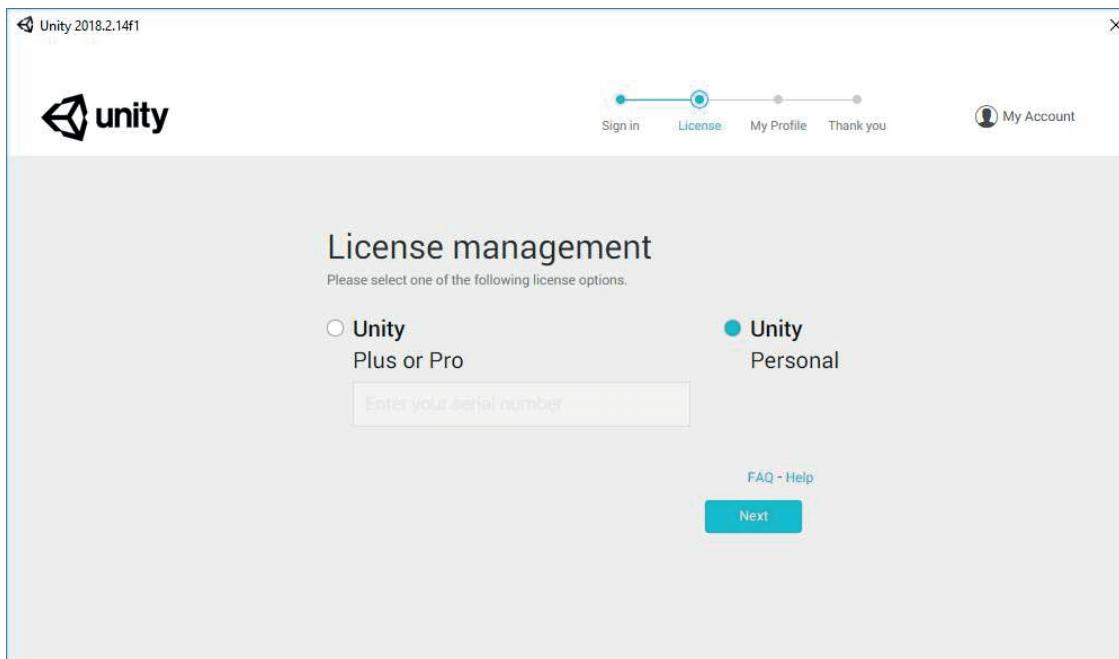
5

Una vez que hayamos creado una cuenta, iniciamos sesión en Unity y confirmamos todos los detalles de su licencia.

6

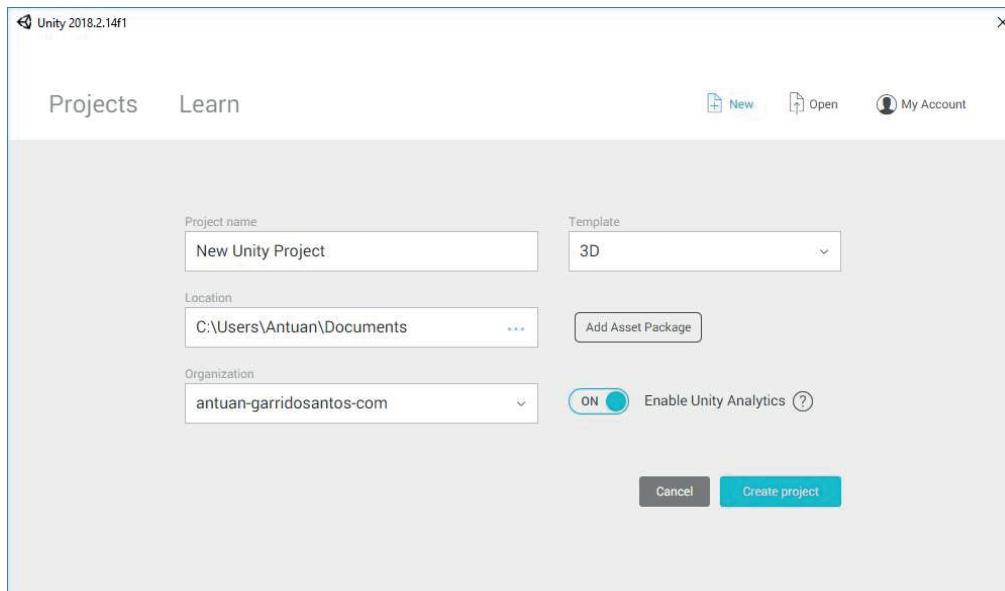
Aparecerá automáticamente el diálogo *proyectos*, que nos permitirá crear un nuevo proyecto.





7

Hacemos clic en el botón **New Project** y vemos que el diálogo cambia de la siguiente manera:



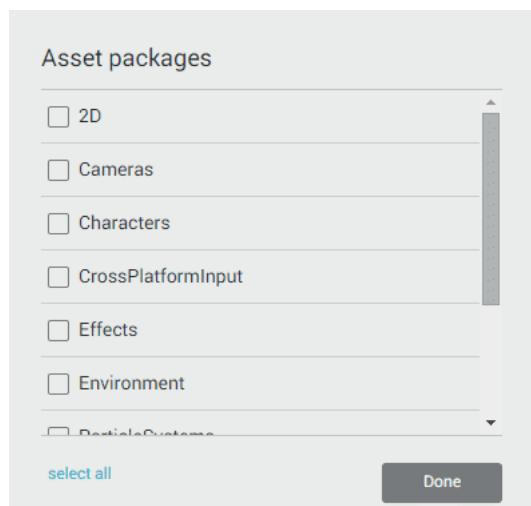
8

Aquí podemos asignar un nombre al proyecto y elegir una ubicación en el disco duro para guardarlo.

Vemos que hay distintas opciones: 3D o 2D, y un interruptor para Unity Analytics. Las opciones 2D o 3D determinan la configuración inicial del editor. Podemos cambiar esta configuración más adelante si decidimos modificar el tipo de juego. Por ahora, seleccionamos 3D.

9

El botón *Add Assets packages* nos permite importar paquetes personalizados en el proyecto.



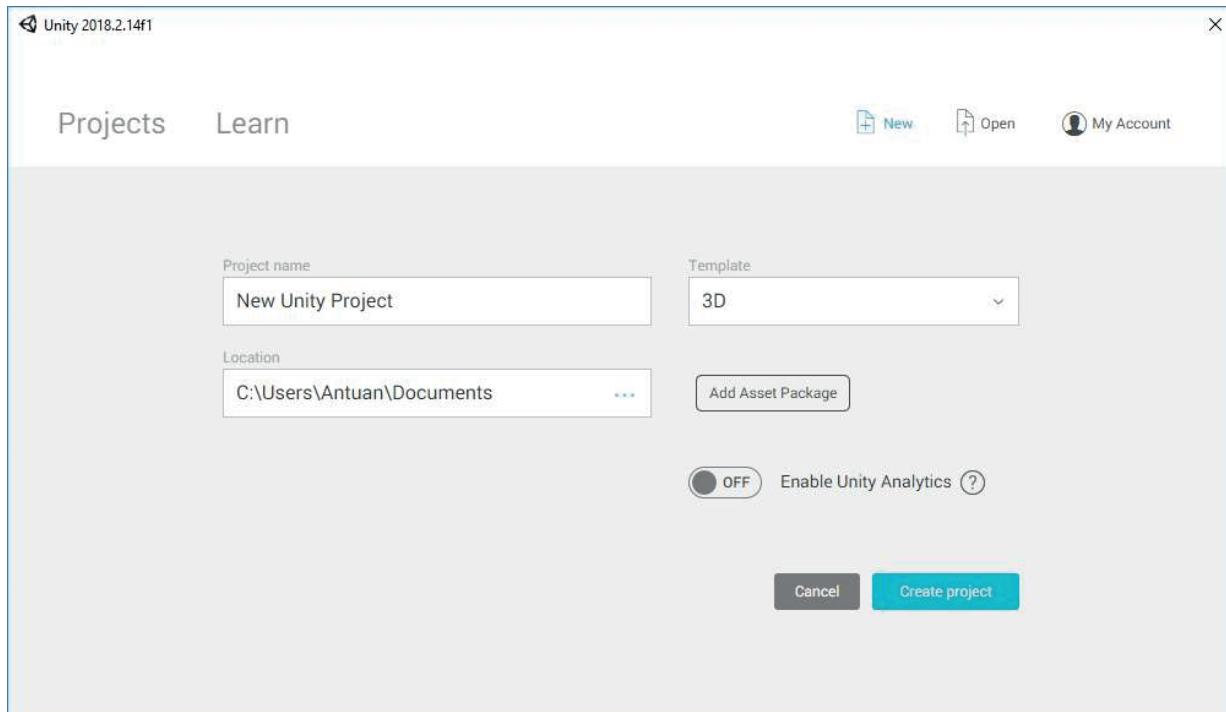
i Recuerda que podemos importar paquetes en cualquier momento del proyecto.

10

Desmarcamos Unity Analytics.

11

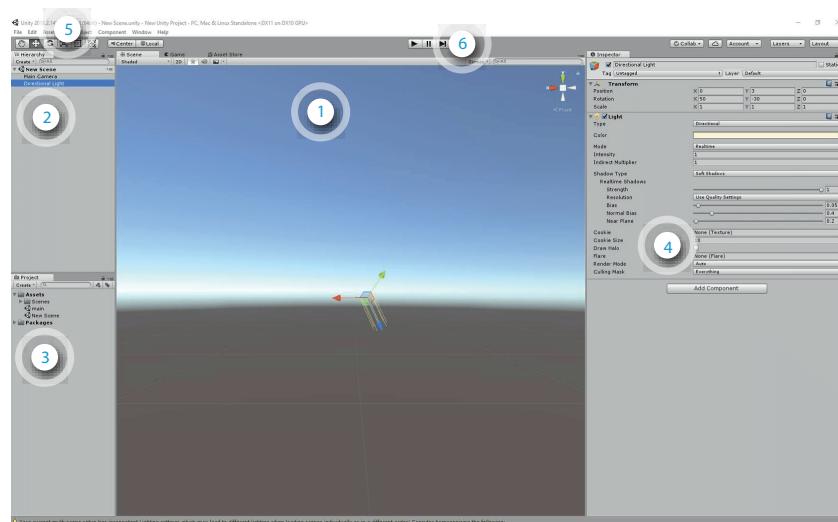
Finalmente, hacemos clic en *Create project* y Unity se abrirá con nuestro primer proyecto.



Área de trabajo

En el siguiente apartado te mostraremos las distintas **vistas y paneles de nuestra área de trabajo en Unity**.

Aquí tienes una captura donde te señalamos los **elementos principales de este área de trabajo**.



1 Vistas de *Escena* y *Juego*.

2 Panel de *Jerarquía*.

3 Panel de *Proyecto*.

4 Panel de *Inspector*.

5 Barra de herramientas.

6 Controles de reproducción.

A continuación los analizaremos uno por uno.

Vistas de *Escena* y *Juego*

1

Vista de *Escena*

Es donde construimos el juego, es decir, donde agregamos todos los modelos, cámaras y otras piezas que conforman el juego. Consiste en una ventana 3D donde podemos ubicar visualmente todos los recursos.

Al probar el juego en Unity, la vista de *Escena* se actualizará con el estado actual del juego, e incluso podremos agregar nuevos elementos sobre la marcha. Cuando detenemos el juego, la vista de *Escena* vuelve al estado original. **Cualquier cambio que hagamos en esta vista mientras juguemos se perderá cuando se detenga el juego.**

2

Vista de *Juego*

Representa la perspectiva del juego del jugador. Aquí es donde podemos jugar y ver cómo funcionan las diferentes mecánicas del juego.

La vista de *Juego* tiene un selector de relación de aspecto, que nos permite cambiar las dimensiones de la vista para que coincidan con la relación de aspecto de una pantalla específica (por ejemplo, 4:3, 16:9, 16:10...), o un tamaño de pantalla del dispositivo como el de iPhone o iPad.



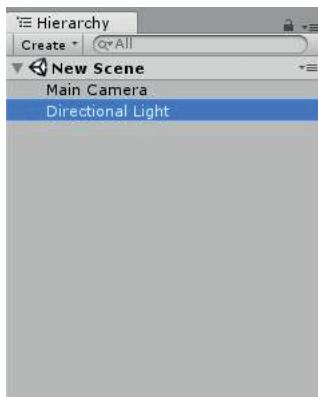
Esto nos permite asegurarnos que se vea bien en todas las relaciones de aspecto y resoluciones, y que no se corte ningún contenido importante. **Podemos agregar una relación de aspecto personalizada utilizando el botón + en la parte inferior del menú desplegable.**

Panel de *Jerarquía*

El panel de *Jerarquía* contiene una lista de todos los *GameObjects* actuales que se usan en el juego.

Como hemos visto, un *GameObject* es un objeto en el juego, un contenedor vacío que podemos personalizar agregando componentes. Los componentes permiten a los *GameObjects* proyectar geometría (desde un simple cubo, hasta modelos 3D más complejos, como torres o monstruos), emitir luz, actuar como cámara, o incluso crear un comportamiento a través de scripts.

Los *GameObjects* también pueden actuar como carpetas que contienen otros *GameObjects*, lo que los hace bastante útiles para organizar la escena.



Una escena representa normalmente un solo nivel del juego, aunque teóricamente podríamos poner todo el juego dentro de una escena. Cualquier *GameObject* que se use en el juego en la escena actual aparecerá en el panel de Jerarquía.

Cada escena nueva comienza con dos *GameObject*s: una cámara principal y una luz direccional.

Recuerda que cuando eliminemos un *GameObject* de la Jerarquía, lo estaremos eliminando también de la escena.

Panel de Proyecto

El panel de *Proyecto* contiene todos los recursos utilizados por el juego.

Podemos organizar los *assets* por carpetas y, cuando deseemos usarlos, simplemente arrastrarlos desde el panel de *Proyecto* al de *Jerarquía*. Alternativamente, podemos arrastrarlos desde el panel de *Proyecto* a la vista de *Escena*. Si arrastramos archivos desde el ordenador a la ventana de *Proyecto*, Unity los importará automáticamente como *assets*.

Unity organiza los *assets* en la ventana de *Proyecto* de la misma manera que el sistema de archivos del sistema operativo.

Podemos realizar cambios directamente en las carpetas de archivos en lugar de hacerlo en la ventana de *Proyecto*, pero es un error que afectará a los *assets* y posiblemente al juego.

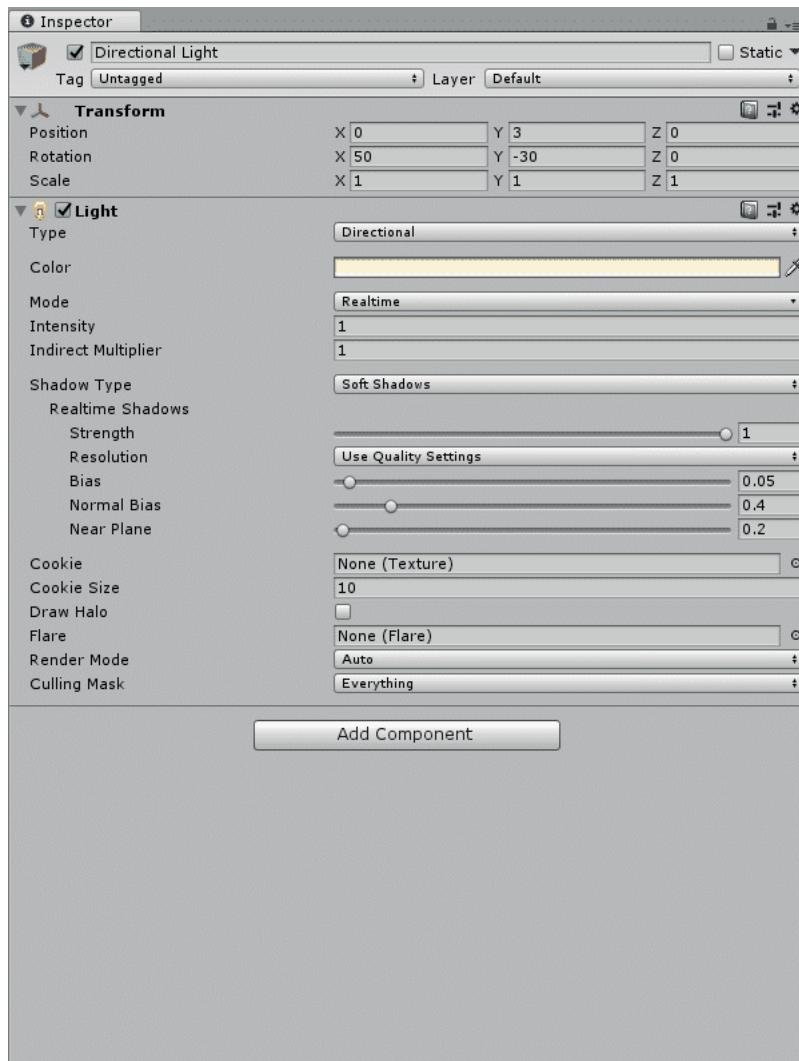


- Unity tiene metadatos en cada asset, por lo que mover assets entre carpetas en el sistema de archivos desvincula los metadatos. Si necesitamos realizar cambios organizativos en los assets, siempre será mejor hacerlo desde el panel de Proyecto.

Panel de Inspector

El panel de *Inspector* nos permite configurar cualquier *GameObject*. Cuando seleccionamos un *GameObject* en el panel de *Jerarquía*, el panel de *Inspector* muestra todos los componentes y las propiedades de ese *GameObject*.

Por ejemplo, una luz tendrá un campo de color junto con un campo de intensidad.



Barra de herramientas

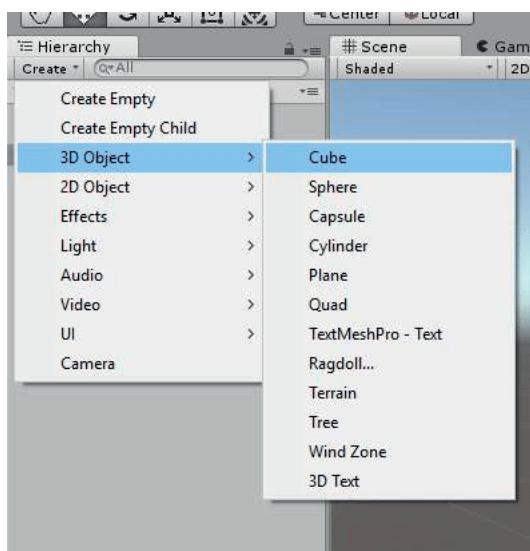
La barra de herramientas sirve para manipular los diversos *GameObjects* en la vista de Escena.



Para ver el efecto y cómo funciona, **vamos a agregar un cubo simple a la escena.**

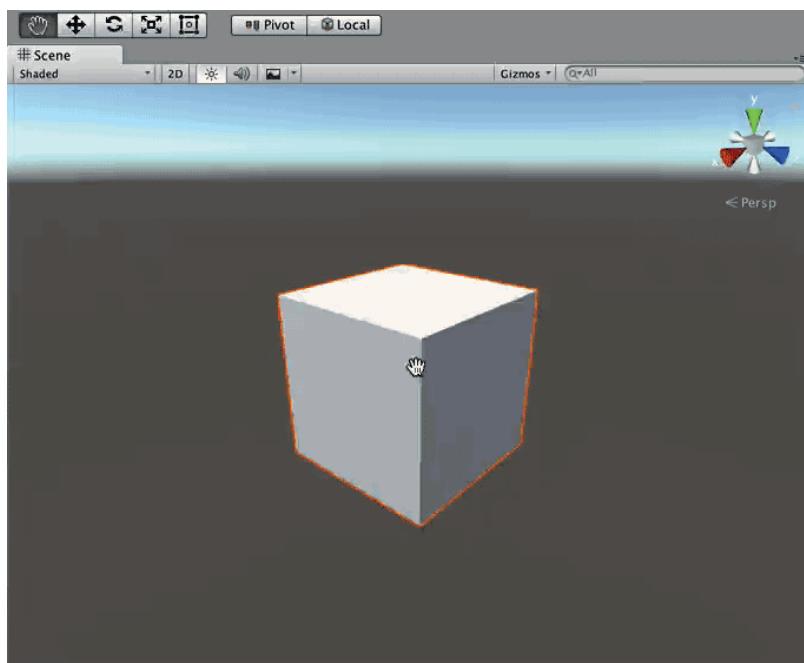
1

Lo haremos desde el botón *Create* del panel de *Jerarquía*.



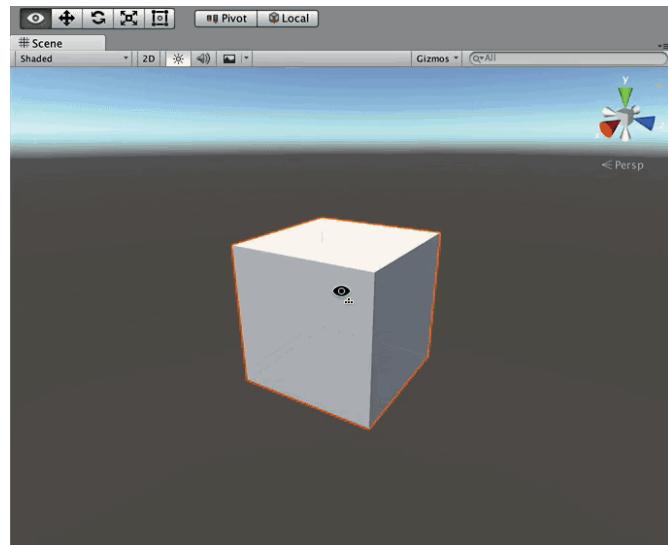
2

La herramienta *Mano* posee un montón de características útiles integradas. La seleccionamos y luego arrastramos sobre la vista *Escena* para recorrer toda la escena.



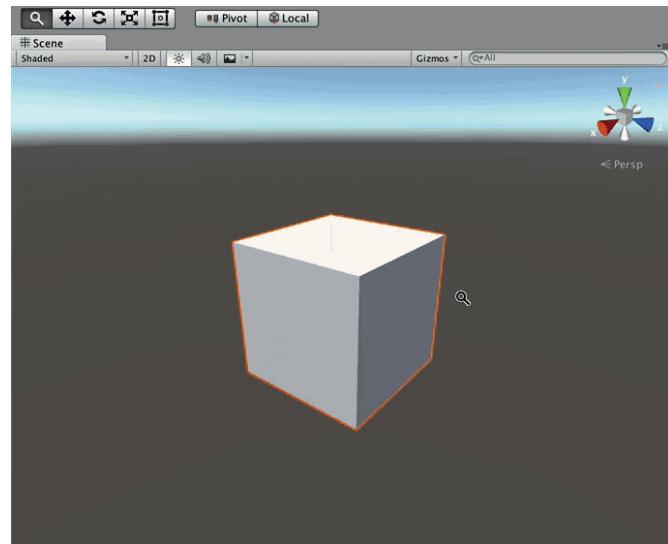
3

Hacemos clic en el botón derecho del ratón y arrastramos para girar la cámara alrededor de la posición actual. Vemos que el icono de la mano se transforma en un ojo.



4

Mantenemos presionadas las teclas Ctrl (o Cmd) y Alt (u Opción), hacemos clic con el botón derecho y **movemos el ratón hacia arriba para acercarnos a la escena**. Moviendo el ratón hacia abajo se aleja. También podemos usar la rueda del ratón.



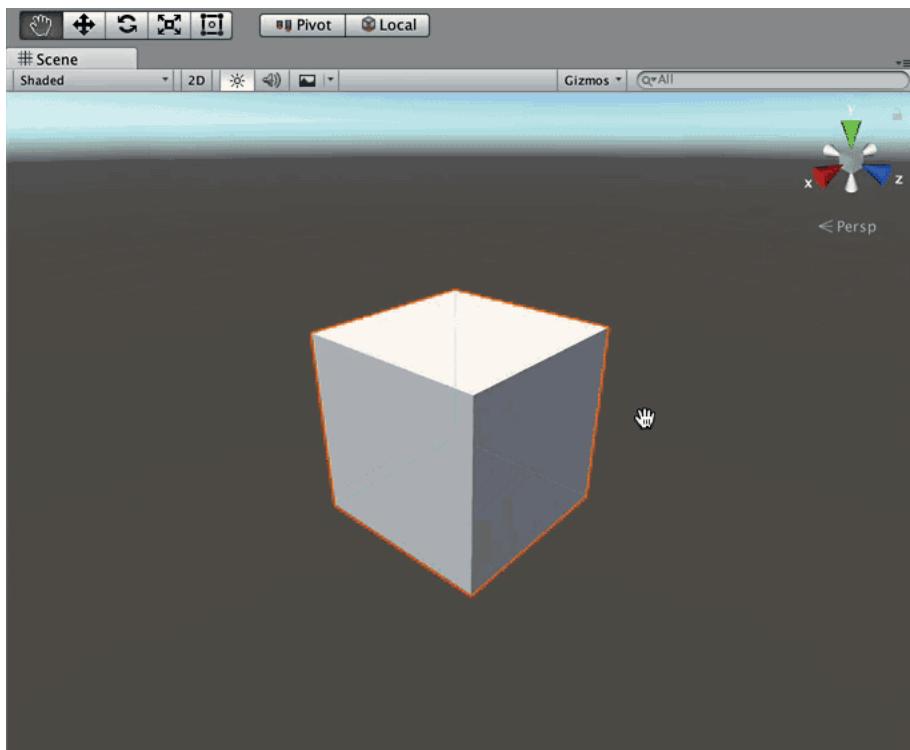
5

Por último, si mantenemos presionado el botón derecho del ratón y usamos las teclas WASD para movernos por la escena, lo haremos como lo haríamos en un juego de acción en primera persona.

Del mismo modo, podemos subir y bajar presionando E o Q respectivamente.

6

Para movernos más rápido, presionamos la tecla Mayúsculas.



Además de estas, en la barra de herramientas hay **numerosas herramientas** más, que te mostraremos en el siguiente apartado.

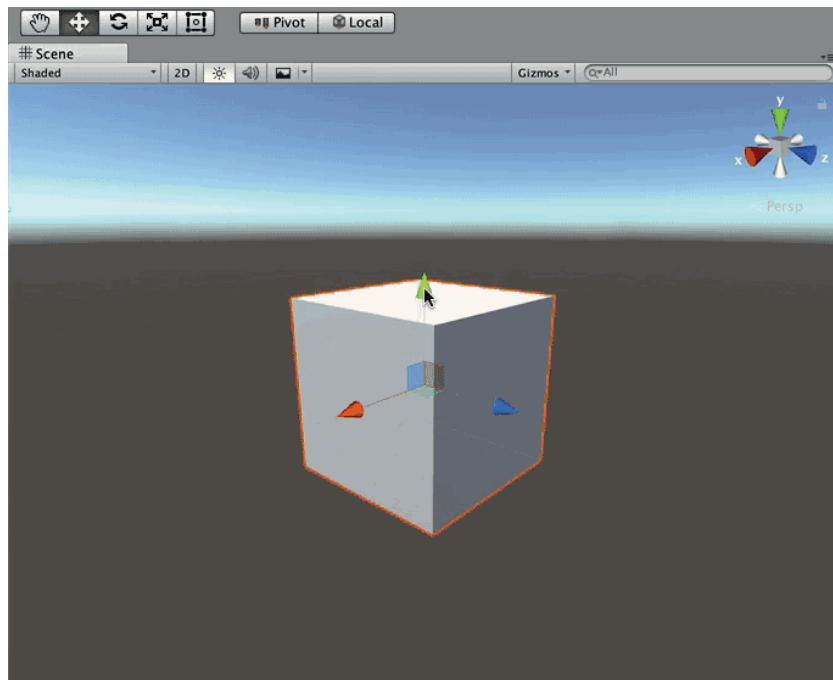
Herramientas

En el apartado anterior te hemos mostrado las **principales vistas y algunas herramientas**, como la herramienta *mano*.

Ahora nos detendremos en **otras herramientas disponibles en la barra de herramientas**, para que aprendas cómo usarlas.

Herramienta *Trasladar*

Cuando seleccionamos un *GameObject* mientras la herramienta *trasladar* está activa, vemos tres flechas de colores saliendo desde el centro. Estas flechas indican los ejes tridimensionales del objeto:



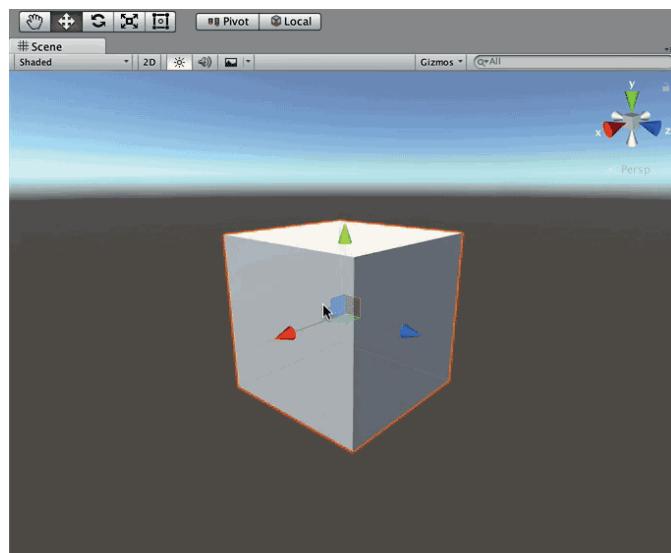
- El eje **x** es la flecha roja.
- El eje **y** es la flecha verde.
- El eje **z** es la flecha azul.

- i** Este grupo de tres flechas de colores se conoce como *gizmo* en Unity. Un *gizmo* es una geometría o textura 3D que proporciona información sobre el *GameObject*.

En este caso, el *gizmo* indica la dirección de cada eje. Es posible agregar nuestros propios *gizmos* para indicar cosas como los puntos de generación, la ubicación de las cámaras o, incluso, la dirección del viento. Lo mejor de los *gizmos* es que solo se muestran en la vista de Escena, y podemos quitarlos cuando ya no sean necesarios.

Hay muchas formas de mover los *GameObjects* con la herramienta *Trasladar*. Podemos seleccionar una de las flechas (que se volverá amarilla cuando esté seleccionada), y luego mover el ratón. En este caso, el *GameObject* solo se moverá en ese eje.

Los tres planos de colores en el centro del *GameObject* nos permiten moverlo en dos ejes a la vez. El color del cuadrado indica en qué plano se moverá. Por ejemplo, un cuadrado verde significa que podemos mover un objeto en el plano y (lo que significa que puede moverse en los ejes x e y):



Herramienta Rotación

La herramienta de rotación nos permite rotar objetos.

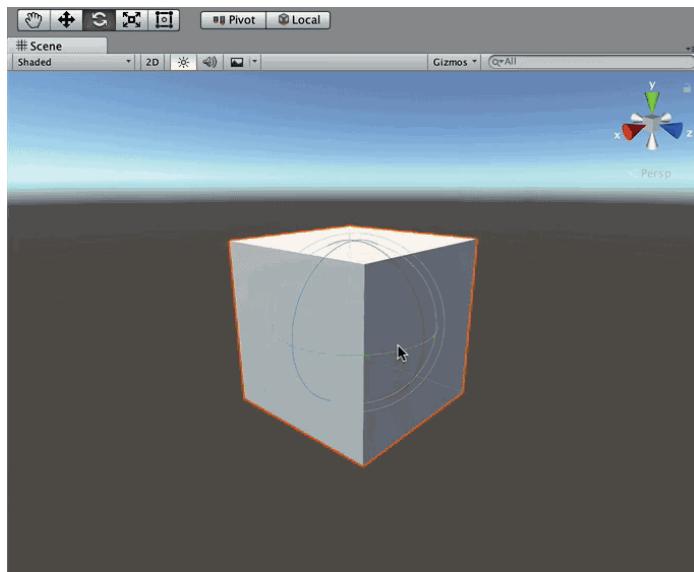
Cuando seleccionamos un objeto para rotar vemos tres líneas esféricas de colores. Estas líneas indican el eje para girar alrededor.

1

Seleccionamos la línea y movemos el ratón para rotar el *GameObject*.

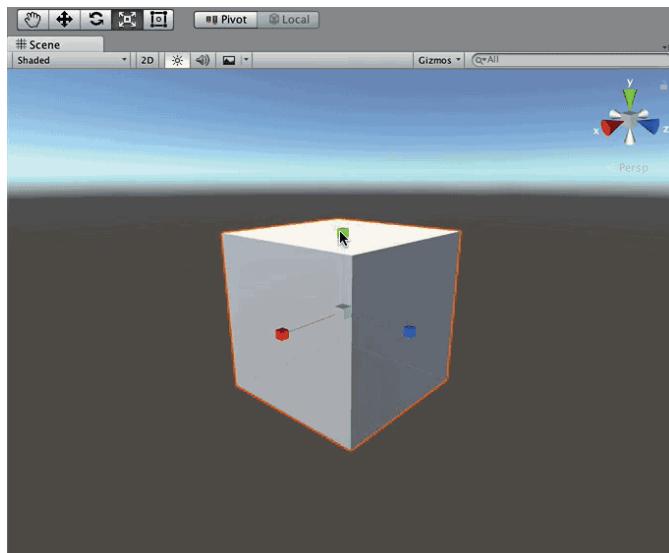
2

Para girar libremente sin estar limitado a un eje, seleccionamos el espacio entre las líneas y movemos el ratón.



Herramienta Escalar

La herramienta escalar nos permite **escalar un GameObject** a lo largo de un solo eje, o proporcionalmente en todos los ejes.



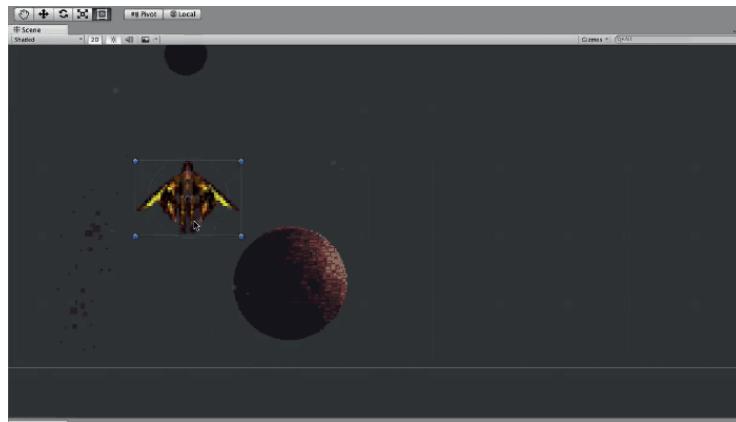
- Para **escalar un objeto a lo largo de un eje**, seleccionamos el *GameObject* y luego arrastramos el extremo cuadrado de una línea de eje. El *GameObject* escalará a lo largo de ese eje.

- Para **escalar todo el GameObject**, seleccionamos su punto central y movemos el ratón hacia adelante o hacia atrás, para escalar el *GameObject* hacia arriba o hacia abajo.

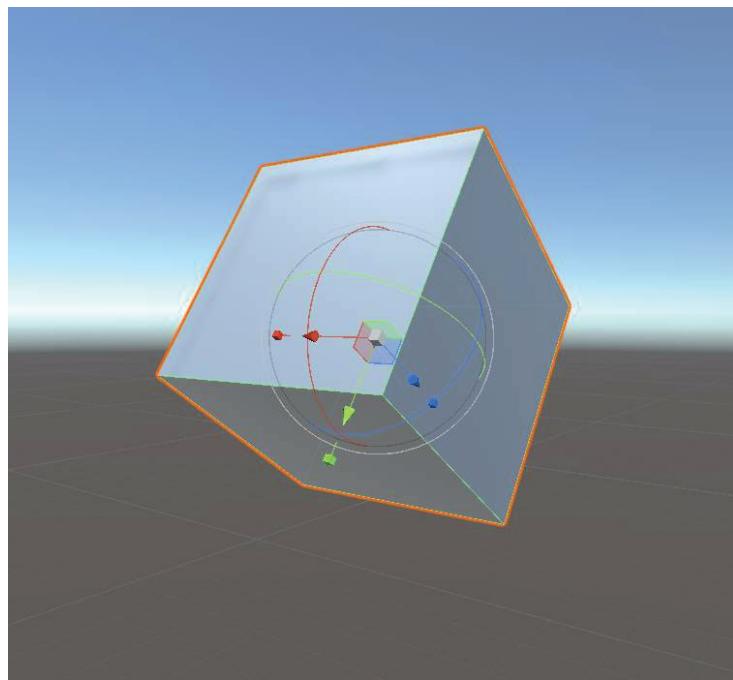
Herramienta Rect

La herramienta Rect se usa principalmente con los *GameObjects* 2D y *GameObjects* de interfaz de usuario (UI), aunque también podemos usarla en *GameObjects* 3D.

La herramienta puede redimensionar, escalar y rotar *assets* 2D y también restablecer el punto de pivote, que es el punto central de rotación del objeto. El punto de pivote está representado por un círculo grande en el *asset*.



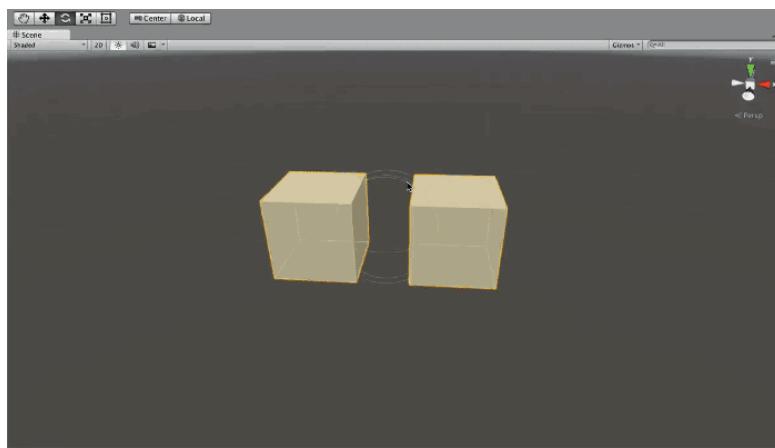
El último botón nos proporciona una visión y uso de los *gizmos* de las tres herramientas anteriores: *trasladar*, *rotar* y *escalar*.



Los comutadores de visualización del *gizmo* pueden ser un poco confusos. De hecho, más que botones son interruptores que controlan la forma en la que colocamos los *gizmos* en la escena.

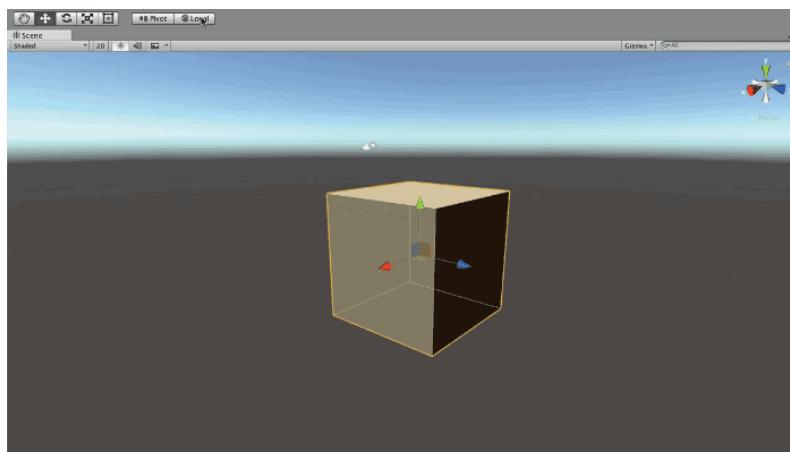
- El primer interruptor cambia entre el modo *Center* y *Pivot*. Cuando está en modo *center*, si seleccionamos dos *GameObjects*, el *gizmo* se colocará en el centro de los dos objetos.
- Girando los *GameObjects* se mueven alrededor del *gizmo* central.

En el modo de *pivot*, cada *GameObject* gira alrededor de su propio punto de giro, como puedes ver:



- El segundo interruptor alterna entre el espacio global y el local. En el modo *global*, estamos manipulando el objeto desde el punto de vista del mundo que lo rodea. El eje x va de izquierda a derecha, el eje y sube y baja, y el eje z avanza y retrocede como de costumbre.

Al cambiar a modo local funcionamos desde el sistema de coordenadas del objeto en sí mismo, y cambian los ejes para que coincidan con los del propio *GameObject*.



Controles de reproducción

Los botones de juego permiten **iniciar y detener el juego**.

1

El primer botón es el botón *Reproducir*. Cuando lo pulsamos, el juego comenzará, como se esperaba. El conjunto de botones se volverá azul para indicar que el juego está en movimiento, y la interfaz de diseño completa se atenuará.

Es muy importante recordar que **cualquier cambio que hagamos sobre la marcha se perderá una vez que detengamos el juego**. Se trata de un error muy común, ejecutar el juego y hacer un montón de cambios, que desaparecen cuando el juego se detiene.

2

El siguiente es el botón *pausa*, que detiene el juego y nos permite hacer modificaciones. Al igual que en el modo de juego, esas modificaciones se perderán una vez que detengamos el juego.

3

Finalmente está el botón *siguiente*, que nos permite recorrer el juego fotograma a fotograma. Es útil cuando queremos observar animaciones, o comprobar el estado de determinados *GameObjects* durante el juego.



Resumen

Hemos terminado la lección, repasemos los puntos más importantes que hemos tratado.

- A lo largo de esta unidad te hemos presentado **Unity**, un motor para desarrollar juegos en Android.
- Hemos aprendido **cómo instalar la versión gratuita de Unity y descuberto las partes fundamentales del área de trabajo del programa**.
- También hemos tenido un primer contacto con el trabajo con los **GameObjects** y los **Assets**.
- Y, para finalizar, hemos profundizado en la **barra de herramientas** y en todas las herramientas que pone a nuestra disposición, que serán de gran importancia para manejar los objetos en nuestra escena.



PROEDUCA