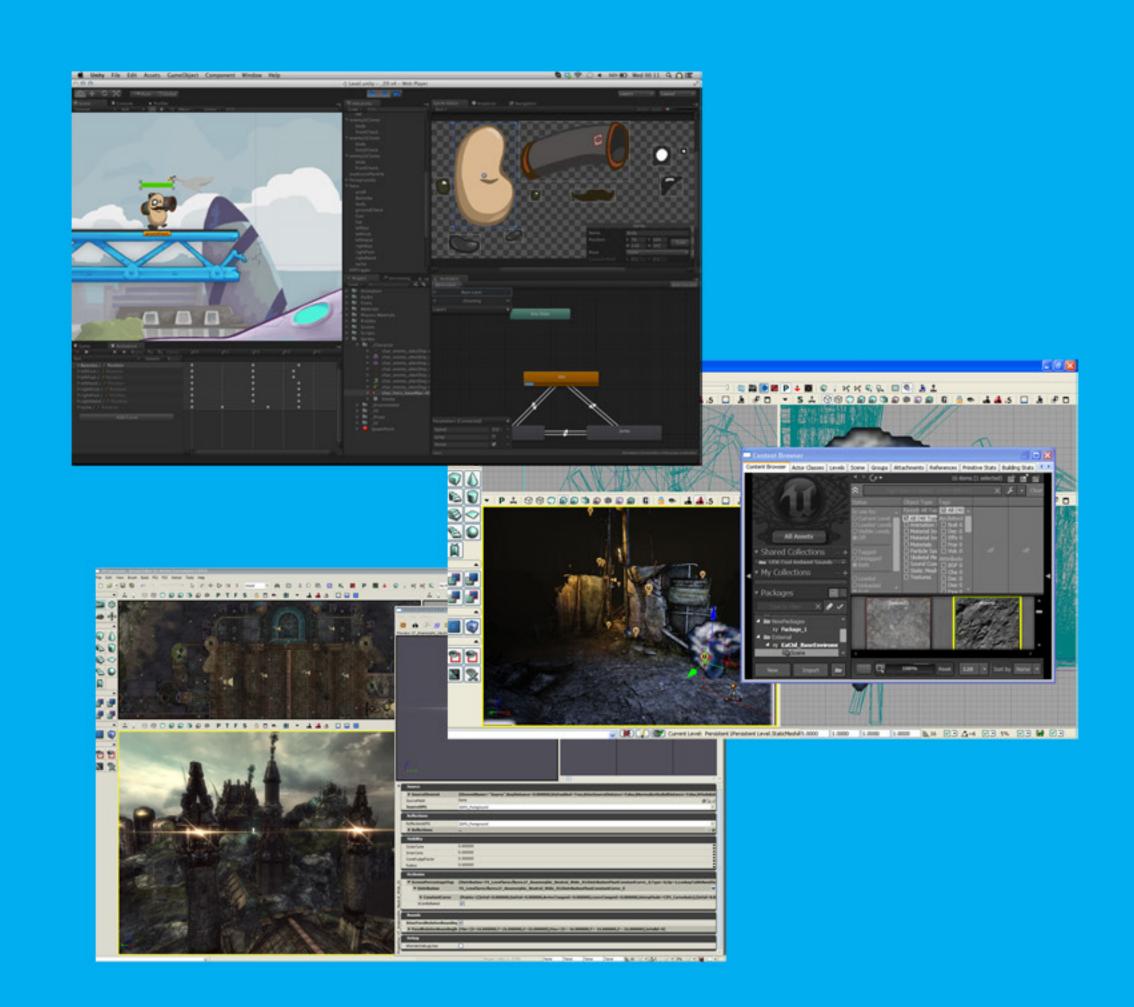


¿Qué es un Game Engine?

Es un programa especialmente diseñano para el desarrollo de videojuegos.

Las empresas de juegos eligen entre hacer sus propios engines o usar alguno de un tercero.



Funcionalidades de un Game Engine

Renderer engine. Para generar gráficos (2D o 3D)

Engine de física o detector de colisiones

Sonido

Animación

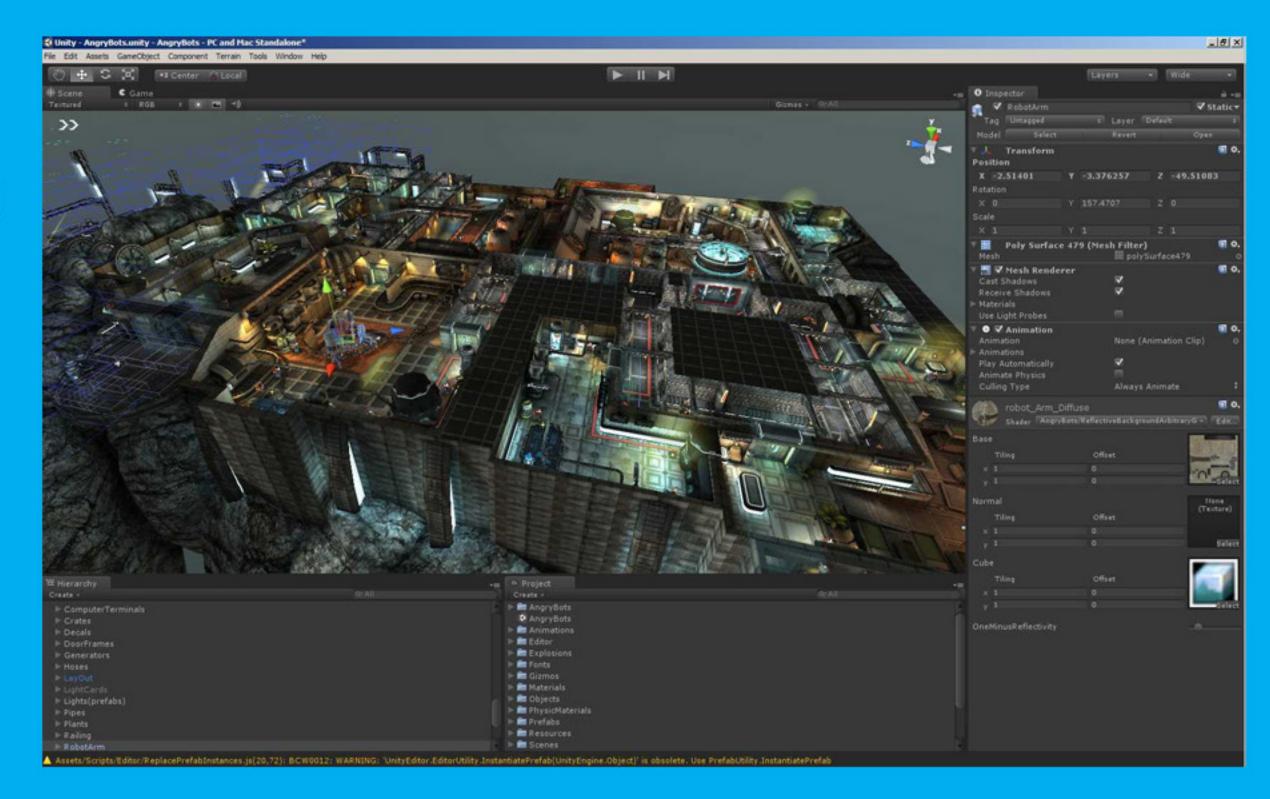
Unity 3D

Desarrollado por Unity Technologies.

Un engine para crear juegos para plugins de la web, consolas, móviles y PCs.

Tiene una versión gratuita (Unity Free) y su versión de paga (Unity Pro)

Unity Pro tiene acceso a más efectos y funcionalidades para los dispositivos.



Características de Unity 3D

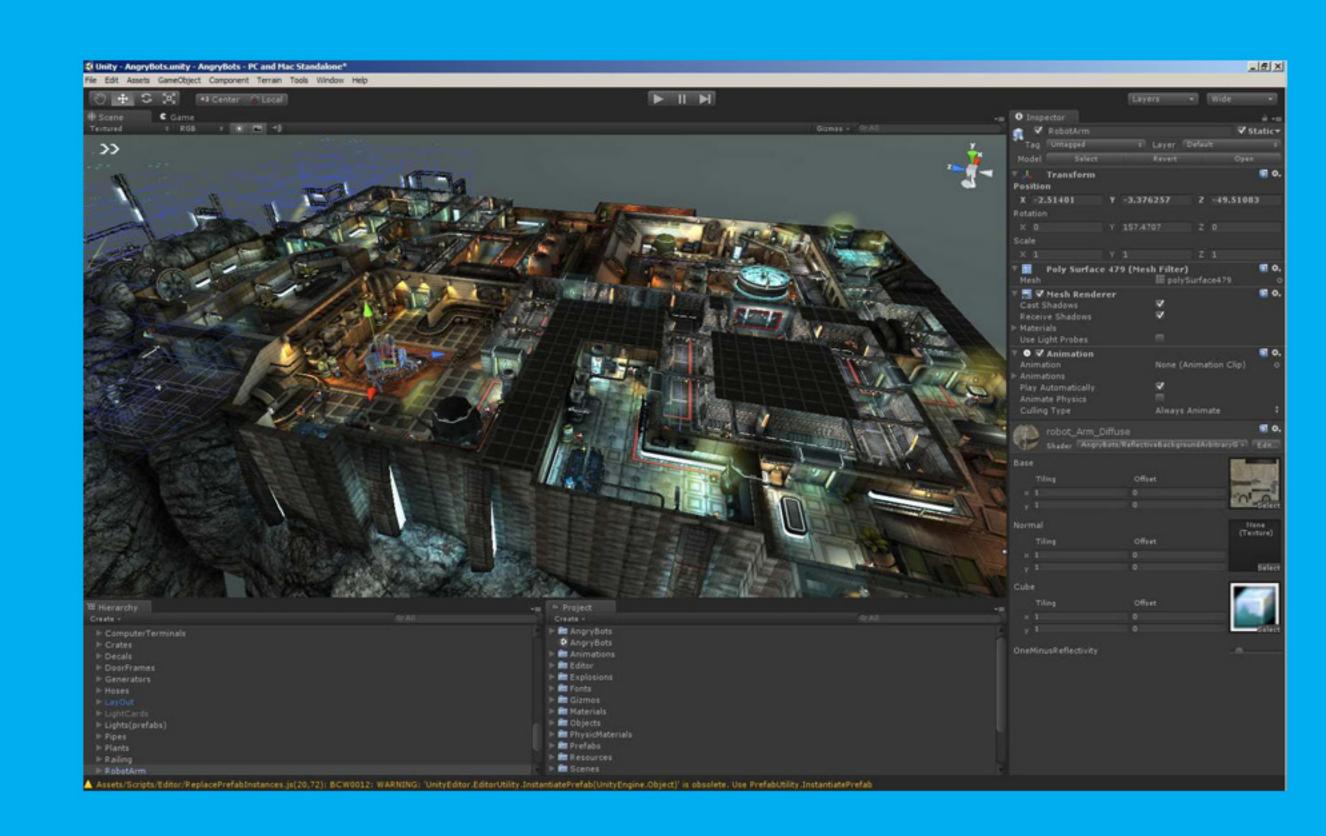
Engine de Rendering

Scripting

Rastreo de Assets

Multi-plataforma

Física



Rendering

Usa Direct3D, OpenGL, OpenGL ES y APIs propietarios dependiendo de la plataforma objetivo.

Se pueden crear efectos de bump mapping, parallax, reflection, sombras dinámicas, etc.

Usa el lenguaje ShaderLab para crear shaders.

Scripting

Creado en Mono

Puede usar scripts de Javascript, C# y Boo

```
Assembly-CSharp - Assets\ScriptGame\scriptPlayer.cs - MonoDevelop-Unity
File Edit View Search Project Build Bun Version Control Tools Window Help
 □ 🗎 🖫 🕼 🦠 🐐 🖟 🛗 📵 🔞 🚇 👰 🖟 🖟 🖟 🖟 Debug
                                                                       ▼ 🧸 🌋 🔗 🐗 🔗 📵 Solution 🕝 💠 — 💠 🚜 🔞 ▼ 📵 🗐 👩 🖽 😢
  E Globals.cs × E GameControl.cs × E GameplayUpdate.cs × E EndGame.cs × E AudioScript.cs × E scriptEnemy.cs × E scriptCell.cs × E ButtonTouch.cs ×
   ScriptPlayer → 🗇 downPress ()
       1 = using UnityEngine;
         using System.Collections;
       4 - public class scriptPlayer : MonoBehaviour (
             bool going, spotReached, north, south, east, west;
              public short direction, steps, speed;
              movePlayer moveP; GameObject gameSFX; GameObject gameBGM; AudioScript sfxScript; AudioScript bgmScript;
              Color colorStart; Color colorEnd; float duration = 0.3f;
              void Start() (// Use this for initialization
               going = false; north = true; south = false; east = false; west = false; spotReached = true;
                speed = Globals.pSpeed; steps = 0; direction = 1; //N = 1, E = 2, S = 3, W = 4
                moveP = GetComponent<movePlayer>();
                gameSFX = GameObject.FindWithTag("Audio"); sfxScript = gameSFX.GetComponent<AudioScript>();
                gameBGM = GameObject.FindWithTag("Music"); bgmScript = gameBGM.GetComponent<AudioScript>();
                colorStart = renderer.material.color; colorEnd = new Color(colorStart.r, colorStart.g, colorStart.b, 0);
              void Update() { if(!Globals.paused){// Update is called once per frame
               //check if still moving to square
                if (going) { //moving to next space
                  if(north) { direction = 1; transform.eulerAngles = new Vector3(0, 0, 0); }
                  if(east) { direction = 2; transform.eulerAngles = new Vector3(0, 0, 270); }
                  if(south) { direction = 3; transform.eulerAngles = new Vector3(0, 0, 180); }
                  if(west) { direction = 4; transform.eulerAngles = new Vector3(0, 0, 90); }
                  spotReached = false; if(steps == 3) sfxScript.sndMove();
                  //move in direction
                     //count steps until in next space
                  if(Globals.readyP) moveP.Move(direction); steps++; if(steps >= (20/speed))( steps = 0; spotReached = true; }
                else{ //let go of button
                 if(!spotReached){ //let go of button, but still moving
                   if (Globals.readyP) moveP.Move(direction); steps++; if (steps >= (20/speed)) { steps = 0; spotReached = true; }
                                                                                                                                        57:59 INS Peedback Task List
```

Rastreo de Assets

Usa PostgreSQL para administrar los assets

Usa PostgreSQL para administrar los assets

Tiene un sistema de audio construido en la librería FMOD

Reproduce video usando Theora códec

Engine de terreno y vegetación

Iluminación global usando Beast

Conexiones para multijugadores usando RakNet

Multi-Plataforma

Windows, Mac OS y Linux

Unity Web Player y Adobe Flash

iOS, Android, Windows Phone 8 y BlackBerry 10

PlayStation 3, Xbox 360, Wii y WiiU

















Física

Para la física, Unity usa el engine de Nvidia, PhysX

Soporta física para simulación de tela sobre objetos irregulares en tiempo real

A partir de Unity 4.3, usa Box2D para la física en juegos 2D



