



Unreal vs. Unity

There are a number of video game engines out there and if you're looking to develop a game, you'll need to choose the right one for your project. The popular engines are the safe bet because they have more users, meaning they are always being updated/developed and have loads of documentation & information available online. Today, we're pitting Unreal vs Unity. That is, Unreal Engine 4 versus Unity.

Do Not Press Start to Skip

If you're thinking of making a video game, there are a lot more steps you need to go through before you get to the game engine. Because of the scope and size, video games are often designed in teams. A [Game Design Doc](#) is used to organize the effort and the vision of the game. Even if you're developing the game by yourself, it's wise to have a GDD. Think of it as an outline or a set of storyboards for the design and development. Later, when you're looking for assets, or once you're in the game engine, you can focus on the work at hand instead of finding a style or inventing gameplay on the spot. So, what is the story? The gameplay? What games are similar? Who is your audience? These are some of the questions you should answer before starting development.

After your GDD is complete, you should design your levels on paper. Also, you'll need to get going on assets to bring into the game engine. In other words, you need the artwork, objects & characters that make up the game. These assets are described in the GDD, so if you have multiple team members they'll know exactly what you're looking for because you've all discussed and documented these items already.

That was a very simple and brief look at the workflow of game developing that takes place before you jump into Unity or UE4. Keep in mind that if you're going to do this solo, it may be smart to design an incredibly simple game, or a single, playable level for your first game engine project. This way you will learn your way around and see some of the challenges and issues you'll face with a larger project.

PIC TUTs

README

As far as video tutorials and documentation both, Unity and Unreal, have a decent library. Unity has a large community because it has had free version of the engine for quite a while. It hasn't quite been a year since UE4 became free. [Aaron Clifford](#) recommends "[Learn to Code by Making Games - the Complete Unity Guide](#)" from udeemy. Aaron added, "The guys who put together the course have built a community of game developers around them, many of whom are eager to answer questions." Personally, he has used both Unreal and Unity, but during [our previous chat](#) Aaron told me that Unity made sense for him because he uses C# in his day job.

I also reached out to [Garret Spenst](#) who said he learned scripting in Unity. He said, "The best part of Unity, and Epic is picking up on this as well, is their community support." In fact, Garret thinks the [Unreal Slackers group](#) is an invaluable resource for UE4 users. He also notes that Youtube and forums are also great ways to find information. Both of the guys I talked to, [Garret](#) and [Aaron](#), said they're happy to help if you want to shoot your Unreal Engine 4 or Unity questions at them.

Strength In Unity

If you're interested in making mobile or 2D games, Unity might be your best choice. In my research & conversations, it appears that UE4 just isn't 2D friendly. That doesn't mean you can't make a 3D game, with a 2D perspective. Garret mentioned [Trine](#) is a great example of that. There is [documentation & examples](#) if you want to give it a try. Though, Unity's got you covered with [solid tutorials on 2D game creation](#), right on their site. That being said, I've got a real curveball for you. Aaron Clifford told me that he's moved to [Construct 2](#) for 2D games and uses Unity for 3D work. It's all a matter of personal taste, right?

Unity games are less intense on hardware compared to Unreal Engine 4 which is why it is currently better suited for mobile games. Again, UE4 falls short in this category. That's not to say that there are no UE4 mobile games, [there are a number listed on the Ureal wiki](#). Plus, [they've got the documentaion you'll need](#) to design mobile games. As I said before, Unity has just been free for longer and the community has pushed the development further in these areas.

Now that UE4 is free for developers and mobile devices are becoming more powerful, things could change. Garret reminded me that [Vulkan](#) is slowly finding its way into the market. This will greatly improve the GPU usage of phones and desktops.

Unreal Artistic Flare

One of the most noticeable things in this research was the way artists gravitate towards Unreal Engine 4. The people in tutorials and other Youtube videos I watched all mention that UE4 is a great tool for artists. Even [Midge Sinnaeve](#) told me that some architectural visualization artists are using UE4 instead of Maya, Blender, C4D or 3ds Max. Obviously, being more hardware intense allows artists to make some beautiful, photorealistic creations. The tools in UE4 for lighting and design will allow you to make a game that looks like it should be on a console system.

PIC Blueprint

Indeed, UE4 is so focused on artwork that Epic has included [Blueprint](#) a node-based scripting interface to program the game visually. You'll have to buy [Playmaker](#) or a similar add-on for Unity, if you want visual scripting.

Game Over. Restart or Continue?

If you're looking to build your masterpiece, you don't need to know what game engine is "the best." You need to find the one you like the most. How do you do that? Garret Spent says, "just download it and start ripping apart their starter content and just go!" He's a good example of finding a personal solution since he started in Unity and found his way to Unreal Engine. Our other friend, Aaron Clifford has used Unreal, but he loves Unity. Furthermore, Aaron has found he prefers Construct 2 over Unity for 2D game projects.

Get familiar with the game engine using the tutorials & documentation and dive in. Put together a plan for a level, or an incredibly small game and open up Unity or UE4 and give it a spin. See the small project through, and if it wasn't the experience you were hoping for, don't continue. Like a video game, end your run and start fresh, just using another engine.

Garret told me he started to feel competent in Unreal Engine after working with it for 3 years. He was also taking university classes at the same time, but the point is that you're not going to know the engine overnight. When you get frustrated, seek help in forums and user groups.

If you've got advice for new users or opinions on Unity or Unreal, we'd love to hear them. Tutorials, resources and other links for user communities would be appreciated as well. Please leave them below in the comments for our readers. Thanks!