

Your Game. Any Screen.













# Microsoft Game Development Guide

September 2017 Edition



# Table of contents

	Introduction to game development for the Universal Windows Platform (UWP)	3
	Game development resources	4
	Concept and planning	10
	Prototype and design	21
	Production	29
	Submitting and publishing your game	33
	Game lifecycle management	35
	Adding Xbox Live to your game	37
	Additional resources	40
	Dream.Build.Play	41

# Welcome to the Windows 10 Game Development Guide!

This guide provides an end-to-end collection of the resources and information you'll need to develop a Universal Windows Platform (UWP) game.



## Introduction to game development for the Universal Windows Platform (UWP)

When you create a Windows 10 game, you have the opportunity to reach millions of players worldwide across phone, PC, and Xbox One. With Xbox on Windows, Xbox Live, cross-device multiplayer, an amazing gaming community, and powerful new features like the Universal Windows Platform (UWP) and DirectX 12, Windows 10 games thrill players of all ages and genres. The new Universal Windows Platform (UWP) delivers compatibility for your game across Windows 10 devices with a common API for phone, PC, and Xbox One, along with tools and options to tailor your game to each device experience.

This guide provides an end-to-end collection of information and resources that will help you as you develop your game. The sections are organized according to the stages of game development, so you'll know where to look for information when you need it.

To get started, the Game development resources section provides a high-level survey of documentation, programs, and other resources that are helpful when creating a game.

This guide will be updated as additional Windows 10 game development resources and material become available.

To get the latest version of this guide, visit [aka.ms/uwpgamedevguide](https://aka.ms/uwpgamedevguide)



# Game development resources

From documentation to developer programs, forums, blogs, and samples, there are many resources available to help you on your game development journey. Here's a roundup of resources to know about as you begin developing your Windows 10 game.

## NOTE

Some features are managed through various programs. This guide covers a broad range of resources, so you may find that some resources are inaccessible depending on the program you are in or your specific development role. Examples are links that resolve to developer, xboxlive.com, forums.xboxlive.com, xdi.xboxlive.com, or the Game Developer Network (GDN). For information about partnering with Microsoft, see Developer Programs.

## Game development documentation

Throughout this guide, you'll find deep links to relevant documentation—organized by task, technology, and stage of game development. To give you a broad view of what's available, here are the main documentation portals for Windows 10 game development.

Windows Dev Center main portal	<a href="https://aka.ms/windows-dev-center">Windows Dev Center</a> <a href="https://aka.ms/windows-dev-center">aka.ms/windows-dev-center</a>
Developing Windows apps	<a href="https://aka.ms/dev-windows-app">Develop Windows apps</a> <a href="https://aka.ms/dev-windows-app">aka.ms/dev-windows-app</a>
Universal Windows Platform app development	<a href="https://aka.ms/how-to-guides-for-windows-10-apps">How-to guides for Windows 10 apps</a> <a href="https://aka.ms/how-to-guides-for-windows-10-apps">aka.ms/how-to-guides-for-windows-10-apps</a>
How-to guides for UWP games	<a href="https://aka.ms/games-and-direct-x">Games and DirectX</a> <a href="https://aka.ms/games-and-direct-x">aka.ms/games-and-direct-x</a>
DirectX reference and overviews	<a href="https://aka.ms/direct-x-graphics-and-gaming">DirectX Graphics and Gaming</a> <a href="https://aka.ms/direct-x-graphics-and-gaming">aka.ms/direct-x-graphics-and-gaming</a>

Azure for gaming	<a href="#">Build and scale your games using Azure</a> <a href="#">aka.ms/build-and-scale-your-games-using-azure</a>
UWP on Xbox One	<a href="#">Building UWP apps on Xbox One</a> <a href="#">aka.ms/building-uwp-apps-on-xbox-one</a>
UWP on HoloLens	<a href="#">Building UWP apps on HoloLens</a> <a href="#">aka.ms/building-uwp-apps-on-holo-lens</a>
Xbox Live documentation	<a href="#">Xbox Live SDK</a> <a href="#">aka.ms/Mixer-interactive-documentation</a>
Xbox One developer documentation (GDN)	<a href="#">Xbox One XDK documentation</a> <a href="#">aka.ms/xbox-one-xdk-documentation</a>
Xbox One developer whitepapers (GDN)	<a href="#">White Papers</a> <a href="#">aka.ms/white-papers</a>
Mixer Interactive documentation	<a href="#">Add interactivity to your game</a> <a href="#">aka.ms/mixer-interactive-documentation</a>

## Windows Dev Center

Registering a developer account on the Windows Dev Center is the first step towards publishing your Windows game. A developer account lets you reserve your game's name and submit free or paid games to the Windows Store for all Windows devices. Use your developer account to manage your game and in-game products, get detailed analytics, and enable services that create great experiences for your players around the world.

Microsoft also offers several developer programs to help you develop and publish Windows games. We recommend seeing if any are right for you before registering for a Dev Center account. For more info, go to Developer programs.

Register a developer account	<a href="#">Ready to sign up?</a> <a href="#">aka.ms/ready-ro-sign-up</a>
------------------------------	--

## Developer programs

Microsoft offers several developer programs to help you develop and publish Windows games. Consider joining a developer program if you want to develop games for Xbox One and integrate Xbox Live features in your game. To publish a game in the Windows Store, you'll also need to create a developer account on Windows Dev Center.

### Xbox Live Creators Program

The Xbox Live Creators Program allows anyone to integrate Xbox Live into their title and publish to Xbox One and Windows 10. There is a simplified certification process and no concept approval is required outside of the standard [Windows Store Policies \(aka.ms/Windows-store-policies\)](https://aka.ms/Windows-store-policies).

You can deploy, design, and publish your game in the Creators Program without a dedicated dev kit, using only retail hardware. To get started, download the [Dev Mode Activation app \(aka.ms/dev-mode-activation-app\)](https://aka.ms/dev-mode-activation-app) on your Xbox One.

If you want access to even more Xbox Live capabilities, dedicated marketing and development support, and the chance to be featured in the main Xbox One store, apply to the [ID@Xbox \(aka.ms/ID-atxbox\)](https://aka.ms/ID-atxbox) program.

---

Xbox Live Creators Program

[Learn more about the Xbox Live Creators Program  
aka.ms/Xbox-live-dev-creators-program](https://aka.ms/Xbox-live-dev-creators-program)

---

### ID@Xbox

The ID@Xbox program helps qualified game developers self-publish on Windows and Xbox One. If you want to develop for Xbox One, or add Xbox Live features like Gamerscore, achievements, and leaderboards to your Windows 10 game, sign up with ID@Xbox. Become an ID@Xbox developer to get the tools and support you need to unleash your creativity and maximize your success. We recommend that you apply to ID@Xbox first before registering for a developer account on Windows Dev Center.

---

ID@Xbox developer program

[Independent Developer Program for Xbox One  
aka.ms/independent-developer-program](https://aka.ms/independent-developer-program)

---

ID@Xbox consumer site

[ID@Xbox  
aka.ms/id-xbox](https://aka.ms/id-xbox)

---

## Xbox tools and middleware

The Xbox Tools and Middleware Program licenses Xbox development kits to professional developers of game tools and middleware. Developers accepted into the program can share and distribute their Xbox XDK technologies to other licensed Xbox developers.

Contact the tools and middleware program [xboxtism@microsoft](mailto:xboxtism@microsoft).

## Game samples

There are many Windows 10 game and app samples available to help you understand Windows 10 gaming features and get a quick start on game development. More samples are developed and published regularly, so don't forget to occasionally check back at sample portals to see what's new. You can also [watch GitHub repos \(aka.ms/github-watch\)](#) to be notified of changes and additions.

Universal Windows Platform app samples	<a href="#">Windows-universal-samples</a> <a href="https://aka.ms/windows-universal-samples">aka.ms/windows-universal-samples</a>
Direct3D 12 graphics samples	<a href="#">DirectX-Graphics-Samples</a> <a href="https://aka.ms/directx-graphics-samples">aka.ms/directx-graphics-samples</a>
Direct3D 11 graphics samples	<a href="#">Directx-sdk-samples</a> <a href="https://aka.ms/directx-sdk-samples">aka.ms/directx-sdk-samples</a>
Direct3D 11 first-person game sample	<a href="#">Create a simple UWP game with DirectX</a> <a href="https://aka.ms/create-a-simple-uwp-game-with-directx">aka.ms/create-a-simple-uwp-game-with-directx</a>
Direct2D custom image effects sample	<a href="#">D2DCustomEffects</a> <a href="https://aka.ms/d2d-custom-effects">aka.ms/d2d-custom-effects</a>
Direct2D gradient mesh sample	<a href="#">D2DGradientMesh</a> <a href="https://aka.ms/d2d-gradient-mesh">aka.ms/d2d-gradient-mesh</a>
Direct2D photo adjustment sample	<a href="#">D2DPhotoAdjustment</a> <a href="https://aka.ms/d2d-photo-adjustment">aka.ms/d2d-photo-adjustment</a>
Xbox Advanced Technology Group public samples	<a href="#">Xbox-ATG-Samples</a> <a href="https://aka.ms/xbox-atg-samples">aka.ms/xbox-atg-samples</a>
Xbox Live samples	<a href="#">Xbox Live Samples</a> <a href="https://aka.ms/XboxLive-Samples">aka.ms/XboxLive-Samples</a>

Xbox One game samples (GDN)	<a href="#">Samples</a> aka.ms/xbox-one-game-samples
Windows game samples (MSDN Code Gallery)	<a href="#">Windows Store game samples</a> aka.ms/windows-store-game-samples
JavaScript and HTML5 game sample	<a href="#">JavaScript and HTML5 touch game sample</a> aka.ms/javascript-and-html5-touch-game-sample

## Developer forums

Developer forums are a great place to ask and answer game development questions and connect with the game development community. Forums can also be fantastic resources for finding existing answers to difficult issues that developers have faced and solved in the past.

Windows apps developer forums	<a href="#">Windows store and apps forums</a> aka.ms/windows-store-and-apps-forums
UWP apps developer forum	<a href="#">Developing Universal Windows Platform apps</a> aka.ms/developing-universal-windows-platform-apps
Desktop applications developer forums	<a href="#">Windows desktop applications forums</a> aka.ms/windows-desktop-applications-forums
DirectX Windows Store games (archived forum posts)	<a href="#">Building Windows Store games with DirectX (archived)</a> aka.ms/building-windows-store-games-with-directx
Windows 10 managed partner developer forums	<a href="#">XBOX Developer Forums: Windows 10</a> aka.ms/xbox-developer-forums-windows-10
DirectX forums	<a href="#">DirectX 12 forum</a> aka.ms/directx-12-forum
Azure platform forums	<a href="#">Azure forum</a> aka.ms/Azure-Platform-Forums
Xbox live forum	<a href="#">Xbox Live development forum</a> aka.ms/XboxLive-Forum



## Developer blogs

Developer blogs are another great resource for the latest information about game development. You'll find posts about new features, implementation details, best practices, architecture background, and more.

---

Building apps for Windows blog	<a href="https://aka.ms/building-apps-for-windows-blog">Building Apps for Windows</a> <a href="https://aka.ms/building-apps-for-windows-blog">aka.ms/building-apps-for-windows-blog</a>
Windows 10 (blog posts)	<a href="https://aka.ms/posts-in-windows-10">Posts in Windows 10</a> <a href="https://aka.ms/posts-in-windows-10">aka.ms/posts-in-windows-10</a>
Visual Studio engineering team blog	<a href="https://aka.ms/the-visual-studio-blog">The Visual Studio Blog</a> <a href="https://aka.ms/the-visual-studio-blog">aka.ms/the-visual-studio-blog</a>
Visual Studio developer tools blogs	<a href="https://aka.ms/developer-tools-blogs">Developer Tools Blogs</a> <a href="https://aka.ms/developer-tools-blogs">aka.ms/developer-tools-blogs</a>
Somasegar's developer tools blog	<a href="https://aka.ms/Somasegars-dev-tools-blog">Somasegar's blog</a> <a href="https://aka.ms/Somasegars-dev-tools-blog">aka.ms/Somasegars-dev-tools-blog</a>
DirectX developer blog	<a href="https://aka.ms/directx-developer-blog">DirectX Developer Blog</a> <a href="https://aka.ms/directx-developer-blog">aka.ms/directx-developer-blog</a>
DirectX 12 introduction (blog post)	<a href="https://aka.ms/direct-x-12">DirectX 12</a> <a href="https://aka.ms/direct-x-12">aka.ms/direct-x-12</a>
Visual C++ tools team blog	<a href="https://aka.ms/visual-c-team-blog">Visual C++ team Blog</a> <a href="https://aka.ms/visual-c-team-blog">aka.ms/visual-c-team-blog</a>
PIX team blog	<a href="https://aka.ms/PIX-team-blog">Performance tuning and debugging for games on Windows and Xbox</a> <a href="https://aka.ms/PIX-team-blog">https://aka.ms/PIX-team-blog</a>
Universal Windows App Deployment team blog	<a href="https://aka.ms/Universal-Windows-app-deployment-team-blog">Build and deploy UWP apps team blog</a> <a href="https://aka.ms/Universal-Windows-app-deployment-team-blog">aka.ms/Universal-Windows-app-deployment-team-blog</a>

---



# Concept and planning

In the concept and planning stage, you're deciding what your game is going to be like and the technologies and tools you'll use to bring it to life.

## Overview of game development technologies

When you start developing a game for the UWP, you have multiple options available for graphics, input, audio, networking, utilities, and libraries.

If you've already decided on all the technologies you'll be using in your game, great! If not, the [Game technologies for UWP apps guide \(aka.ms/game-tech-for-uwp-apps-guide\)](#) is an excellent overview of many of the technologies available, and is highly recommended reading to help you understand the options and how they fit together.

---

Survey of UWP game technologies

[Game technologies for UWP apps](#)  
[aka.ms/survey-of-uwp-game-technologies](https://aka.ms/survey-of-uwp-game-technologies)

---

These three GDC 2015 videos give a good overview of Windows 10 game development and the Windows 10 gaming experience.

---

Overview of Windows 10 game development  
(video)

[Developing Games for Windows 10](#)  
[aka.ms/developing-games-for-windows-10](https://aka.ms/developing-games-for-windows-10)

---

Windows 10 gaming experience (video)

[Gaming Consumer Experience on Windows 10](#)  
[aka.ms/gaming-consumer-experience-on-windows-10](https://aka.ms/gaming-consumer-experience-on-windows-10)

---

Gaming across the Microsoft ecosystem (video)

[The Future of Gaming Across the Microsoft Ecosystem](#)  
[aka.ms/the-future-of-gaming-across-the-microsoft](https://aka.ms/the-future-of-gaming-across-the-microsoft)

---

## Game planning

These are some high-level concept and planning topics to consider when planning for your game.

---

Make your game accessible	<a href="#">Accessibility for games</a> <a href="https://aka.ms/accessibility-for-games">aka.ms/accessibility-for-games</a>
Build games using cloud	<a href="#">Cloud for games</a> <a href="https://aka.ms/cloud-for-games">aka.ms/cloud-for-games</a>
Monetize your game	<a href="#">Monetization for games</a> <a href="https://aka.ms/monetization-for-games">aka.ms/monetization-for-games</a>

---

## Choosing your graphics technology and programming language

There are several programming languages and graphics technologies available for use in Windows 10 games. The path you take depends on the type of game you're developing, the experience and preferences of your development studio, and specific feature requirements of your game. Will you use C#, C++, or JavaScript DirectX, XAML, or HTML5?

### DirectX

Microsoft DirectX is the choice to make for the highest-performance 2D and 3D graphics and multimedia.

Direct3D 12, new in Windows 10, brings the power of a console-like API and is faster and more efficient than ever before. Your game can fully utilize modern graphics hardware and feature more objects, richer scenes, and enhanced effects. Direct3D 12 delivers optimized graphics on Windows 10 PCs and Xbox One. If you want to use the familiar graphics pipeline of Direct3D 11, you'll still benefit from the new rendering and optimization features added to Direct3D 11.3. And, if you're a tried-and-true desktop Windows API developer with roots in Win32, you'll still have that option in Windows 10.

The extensive features and deep platform integration of DirectX provide the power and performance needed by the most demanding games.

---

How-to guides for DirectX games	<a href="#">Games and DirectX</a> <a href="https://aka.ms/games-and-directx">aka.ms/games-and-directx</a>
DirectX overviews and reference	<a href="#">DirectX Graphics and Gaming</a> <a href="https://aka.ms/directx-graphics-and-gaming">aka.ms/directx-graphics-and-gaming</a>

---

---

Direct3D 12 programming guide and reference

[Direct3D 12 Graphics](#)  
[aka.ms/direct3d-12-graphics](https://aka.ms/direct3d-12-graphics)

---

Graphics and DirectX 12 development videos  
(YouTube channel)

[Microsoft DirectX 12 and Graphics Education](#)  
[aka.ms/microsoft-directx-12-and-graphics-education](https://aka.ms/microsoft-directx-12-and-graphics-education)

---

## XAML

XAML is an easy-to-use declarative UI language with convenient features like animations, storyboards, data binding, scalable vector-based graphics, dynamic resizing, and scene graphs. XAML works great for game UI, menus, sprites, and 2D graphics. To make UI layout easy, XAML is compatible with design and development tools like Expression Blend and Microsoft Visual Studio. XAML is commonly used with C#, but C++ is also a good choice if that's your preferred language or if your game has high CPU demands.

---

XAML platform overview

[XAML platform](#)  
[aka.ms/xaml-platform](https://aka.ms/xaml-platform)

---

XAML UI and controls

[Controls, layouts, and text](#)  
[aka.ms/controls-layouts-text](https://aka.ms/controls-layouts-text)

---

## HTML 5

HyperText Markup Language (HTML) is a common UI markup language used for web pages, apps, and rich clients. Windows games can use HTML5 as a full-featured presentation layer with the familiar features of HTML, access to the Universal Windows Platform, and support for modern web features like AppCache, Web Workers, canvas, drag-and-drop, asynchronous programming, and SVG. Behind the scenes, HTML rendering takes advantage of the power of DirectX hardware acceleration, so you can still get the performance benefits of DirectX without writing any extra code. HTML5 is a good choice if you are proficient with web development, porting a web game, or want to use language and graphics layers that can be easier to approach than the other choices. HTML5 is used with JavaScript, but can also call into components created with C# or C++/CX.

---

HTML5 and Document Object Model information

[HTML and DOM reference](#)  
[aka.ms/html-and-dom-reference](https://aka.ms/html-and-dom-reference)

---

The HTML5 W3C Recommendation

[HTML5](#)  
[aka.ms/html-5](https://aka.ms/html-5)

---

## Combining presentation technologies

The Microsoft DirectX Graphics Infrastructure (DXGI) provides interop and compatibility across multiple graphics technologies. For high-performance graphics, you can combine XAML and DirectX, using XAML for menus and other simple UI, and DirectX for rendering complex 2D and 3D scenes. DXGI also provides compatibility between Direct2D, Direct3D, DirectWrite, DirectCompute, and the Microsoft Media Foundation.

---

DirectX Graphics Infrastructure programming guide and reference	<a href="#">DXGI</a> <a href="#">aka.ms/dxgi</a>
Combining DirectX and XAML	<a href="#">DirectX and XAML interop</a> <a href="#">aka.ms/directx-and-xaml-interop</a>

---

## C++

C++/CX is a high-performance, low-overhead language that provides the powerful combination of speed, compatibility, and platform access. C++/CX makes it easy to use all of the great gaming features in Windows 10, including DirectX and Xbox Live. You can also reuse existing C++ code and libraries. C++/CX creates fast, native code that doesn't incur the overhead of garbage collection, so your game can have great performance and low power consumption, which leads to longer battery life. Use C++/CX with DirectX or XAML, or create a game that uses a combination of both.

---

C++/CX reference and overviews	<a href="#">Visual C++ Language Reference (C++/CX)</a> <a href="#">aka.ms/visual-c-language-reference</a>
Visual C++ programming guide and reference	<a href="#">Visual C++ in Visual Studio 2015</a> <a href="#">aka.ms/visual-c-visual-studio-2015</a>

---

## C#

C# (pronounced "C sharp") is a modern, innovative language that is simple, powerful, type-safe, and object-oriented. C# enables rapid development while retaining the familiarity and expressiveness of C-style languages. Though easy to use, C# has numerous advanced language features like polymorphism, delegates, lambdas, closures, iterator methods, covariance, and Language-Integrated Query (LINQ) expressions. C# is an excellent choice if you are targeting XAML, want to get a quick start developing your game, or have previous C# experience. C# is used primarily with XAML, so if you want to use DirectX, choose C++ instead, or write part of your game as a C++ component that interacts with DirectX. Or, consider [Win2D \(aka.ms/win2d\)](#), an immediate-mode Direct2D graphics library for C# and C++.

---

C# programming guide and reference	<a href="#">C# language reference</a> <a href="#">aka.ms/c-language-reference</a>
------------------------------------	--

---

## JavaScript

JavaScript is a dynamic scripting language widely used for modern web and rich client applications.

Windows JavaScript apps can access the powerful features of the Universal Windows Platform in an easy, intuitive way—as methods and properties of object-oriented JavaScript classes. JavaScript is a good choice for your game if you're coming from a web development environment, are already familiar with JavaScript, or want to use HTML5, CSS, WinJS, or JavaScript libraries. If you're targeting DirectX or XAML, choose C# or C++/CX instead.

---

JavaScript and Windows Runtime reference

[JavaScript reference](#)  
[aka.ms/javascript-reference](https://aka.ms/javascript-reference)

---

## Use Windows Runtime Components to combine languages

With the Universal Windows Platform, it's easy to combine components written in different languages. Create Windows Runtime Components in C++, C#, or Visual Basic, and then call into them from JavaScript, C#, C++, or Visual Basic. This is a great way to program portions of your game in the language of your choice. Components also let you consume external libraries that are only available in a particular language, as well as use legacy code you've already written.

---

How to create Windows Runtime Components

[Creating Windows Runtime Components](#)  
[aka.ms/How-to-create-Windows-runtime-components](https://aka.ms/How-to-create-Windows-runtime-components)

---

## Which version of DirectX should your game use?

If you are choosing DirectX for your game, you'll need to decide which version to use: Microsoft Direct3D 12 or Microsoft Direct3D 11.

Direct3D 12, new in Windows 10, brings the power of a console-like API and is faster and more efficient than ever before. Your game can fully utilize modern graphics hardware and feature more objects, richer scenes, and enhanced effects. Direct3D 12 delivers optimized graphics on Windows 10 PCs and Xbox One. Since Direct3D 12 works at a very low level, it is able to give an expert graphics development team or an experienced DirectX 11 development team all the control they need to maximize graphics optimization.

Direct3D 11.3 is a low-level graphics API that uses the familiar Direct3D programming model and handles more of the complexity involved in GPU rendering for you. It is also supported in Windows 10 and Xbox One. If you have an existing engine written in Direct3D 11, and you're not quite ready to make the jump to Direct3D 12, you can use Direct3D 11 on 12 to achieve some performance improvements. Versions 11.3+ contain the new rendering and optimization features that are also enabled in Direct3D 12.

---

Choosing Direct3D 12 or Direct3D 11	<a href="#">What is Direct3D 12?</a> <a href="#">aka.ms/what-is-direct3d-12</a>
Overview of Direct3D 11	<a href="#">Direct3D 11 Graphics</a> <a href="#">aka.ms/direct3d-11-graphics</a>
Overview of Direct3D 11 on 12	<a href="#">Direct3D 11 on 12</a> <a href="#">aka.ms/direct3d-11-on-12</a>

---

## Bridges, game engines, and middleware

Depending on the needs of your game, using bridges, game engines, or middleware can save development and testing time and resources. Here are some overview and resources for bridges, game engines, and middleware to help you decide if any are right for you.

---

Game Development with Middleware (video)	<a href="#">Accelerating Windows Store Game Development with Middleware</a> <a href="#">aka.ms/accelerating-windows-store-game-development</a>
Introduction to game middleware (blog post)	<a href="#">Game Development Middleware - What is it? Do I need it?</a> <a href="#">aka.ms/Intro-game-middleware</a>

---

## Universal Windows Platform Bridges

Universal Windows Platform Bridges are technologies that bring your existing app or game over to the UWP. Bridges are a great way to get a quick start on UWP game development.

UWP bridges	<a href="#">Bring your code to Windows</a> <a href="#">aka.ms/uwpgdg_bring-your-code-to-windows</a>
Windows Bridge for iOS	<a href="#">Bring your iOS apps to Windows</a> <a href="#">aka.ms/Windows-Bridge-IOS</a>
Windows Bridge for desktop applications (.NET and Win32)	<a href="#">Convert your desktop application to a UWP app</a> <a href="#">aka.ms/uwpgdg_convert-your-desktop-application</a>

## Unity

Unity offers a platform for creating beautiful and engaging 2D, 3D, VR, and AR games and apps. It enables you to realize your creative vision fast and delivers your content to virtually any media or device.

Beginning with Unity 5.4, Unity supports DirectX 12 development.

The Unity game engine	<a href="#">Unity - Game Engine</a> <a href="#">aka.ms/uwpgdg_unity-game-engine</a>
Get Unity	<a href="#">Get Unity</a> <a href="#">aka.ms/uwpgdg_get-unity</a>
Universal Windows Platform app support in Unity 5.2 or later (blog post)	<a href="#">Windows 10 Universal Platform apps in Unity 5.2</a> <a href="#">aka.ms/uwpgdg_windows-10-universal-platform-apps</a>
Unity documentation for Windows	<a href="#">Unity Manual/Windows</a> <a href="#">aka.ms/uwpgdg_unity-manual-windows</a>
How to add interactivity to your game using Mixer Interactive	<a href="#">Getting started guide</a> <a href="#">aka.ms/How-to-interactivity-game-MixerInteractive</a>
Mixer SDK for Unity	<a href="#">Mixer Unity plugin</a> <a href="#">aka.ms/Mixer-SDK-for-Unity</a>
Mixer SDK for Unity reference documentation	<a href="#">API reference for Mixer Unity plugin</a> <a href="#">aka.ms/Mixer-SDK-Unity-ref-documentation</a>
Publish your Unity game to Windows Store	<a href="#">Porting guide</a> <a href="#">aka.ms/uwpgdg_porting-guide</a>



---

Publish your Unity game as a Universal Windows Platform app (video)	<a href="https://aka.ms/uwpgdg_how-to-publish-your-unity-game-as-uwp">How to publish your Unity game as a UWP app aka.ms/uwpgdg_how-to-publish-your-unity-game-as-uwp</a>
Use Unity to make Windows games and apps (video)	<a href="https://aka.ms/uwpgdg_making-windows-games-and-apps-with-unity">Making Windows games and apps with Unity aka.ms/uwpgdg_making-windows-games-and-apps-with-unity</a>
Unity game development using Visual Studio (video series)	<a href="https://aka.ms/uwpgdg_using-unity-with-visual-studio">Using Unity with Visual Studio 2015 aka.ms/uwpgdg_using-unity-with-visual-studio</a>

---

## Havok

Havok's modular suite of tools and technologies helps game creators reach new levels of interactivity and immersion. Havok enables highly realistic physics, interactive simulations, and stunning cinematics. Version 2015.1 and higher officially support UWP in Visual Studio 2015 on x86, 64-bit, and ARM.

---

Havok website	<a href="https://aka.ms/uwpgdg_havok">Havok aka.ms/uwpgdg_havok</a>
Havok tool suite	<a href="https://aka.ms/uwpgdg_havok-product-overview">Havok Product Overview aka.ms/uwpgdg_havok-product-overview</a>
Havok support forums	<a href="https://aka.ms/uwpgdg_havok-support">Havok aka.ms/uwpgdg_havok-support</a>

---

## MonoGame

MonoGame is an open source, cross-platform game development framework originally based on Microsoft's XNA Framework 4.0. Monogame currently supports Windows, Windows Phone, and Xbox, as well as Linux, macOS, iOS, Android, and several other platforms.

---

MonoGame	<a href="#">MonoGame website</a> <a href="#">aka.ms/uwpgdg_monogame-website</a>
MonoGame Documentation	<a href="#">MonoGame Documentation (latest)</a> <a href="#">aka.ms/uwpgdg_monogame-documentation</a>
Monogame Downloads	<a href="#">Download releases, development builds, and source code from the MonoGame website, or get the latest release via NuGet.</a> <a href="#">aka.ms/uwpgdg_download-releases-development-builds</a> <a href="#">aka.ms/uwpgdg_get-the-latest-release-via-nuget</a>

---

## Cocos2d

Cocos2d-X is a cross-platform open source game development engine and tools suite that supports building UWP games. Beginning with version 3, 3D features are being added as well.

---

Cocos2d-x	<a href="#">What is Cocos2d-X?</a> <a href="#">aka.ms/uwpgdg_what-is-cocos2d-x</a>
Cocos2d-x programmer's guide	<a href="#">Cocos2d-x Programmers Guide v3.12</a> <a href="#">aka.ms/uwpgdg_cocos2d-x-programmers-guide-v3.12</a>
Cocos2d-x on Windows 10 (blog post)	<a href="#">Running Cocos2d-x on Windows 10</a> <a href="#">aka.ms/uwpgdg_running-cocos2d-x-on-windows-10</a>
Cocos2d-x Windows Store games (video)	<a href="#">Build a Game with Cocos2d-x for Windows Devices</a> <a href="#">aka.ms/Cocos2d-x-Windows-store-games</a>

---

## Unreal Engine

Unreal Engine 4 is a complete suite of game development tools for all types of games and developers. For the most demanding console and PC games, Unreal Engine is used by game developers worldwide.

---

Unreal Engine overview

[Unreal Engine 4](#)

[aka.ms/uwpgdg\\_unreal-engine-4](https://aka.ms/uwpgdg_unreal-engine-4)

---

## BabylonJS

BabylonJS is a complete JavaScript framework for building 3D games with HTML5, WebGL, and Web Audio.

---

BabylonJS

[BabylonJS](#)

[aka.ms/uwpgdg\\_babylonjs](https://aka.ms/uwpgdg_babylonjs)

---

WebGL 3D with HTML5 and BabylonJS  
(video series)

[Learning WebGL 3D and BabylonJS](#)

[aka.ms/uwpgdg\\_learning-webgl-3d-and-babylonjs](https://aka.ms/uwpgdg_learning-webgl-3d-and-babylonjs)

---

Building a cross-platform WebGL game  
with BabylonJS

[Use BabylonJS to develop a cross-platform game](#)

[aka.ms/uwpgdg\\_use-babylonjs-to-develop-cross-platform](https://aka.ms/uwpgdg_use-babylonjs-to-develop-cross-platform)

---

## Middleware and partners

There are many other middleware and engine partners that can provide solutions, depending on your game development needs.

---

Windows Dev Center partners

[Dev Center Partners](#)

[aka.ms/uwpgdg\\_dev-center-partners](https://aka.ms/uwpgdg_dev-center-partners)

---

## Porting your game

If you have an existing game, there are many resources and guides available to help you quickly bring your game to the UWP. To jump-start your porting efforts, you might also consider using a [Universal Windows Platform Bridge \(aka.ms/uwpgdg\\_universal-windows-platform-bridge\)](#).

Porting a Windows 8 app to a Universal Windows Platform app	<a href="#">Move from Windows Runtime 8.x to UWP</a> <a href="#">aka.ms/uwpgdg_move-from-windows-runtime-8.x-to-uwp</a>
Porting a Windows 8 app to a Universal Windows Platform app (video)	<a href="#">Porting 8.1 Apps to Windows 10</a> <a href="#">aka.ms/uwpgdg_porting-8.1-apps-to-windows-10</a>
Porting an iOS app to a Universal Windows Platform app	<a href="#">Move from iOS to UWP</a> <a href="#">aka.ms/uwpgdg_move-from-ios-to-uwp</a>
Porting a Silverlight app to a Universal Windows Platform app	<a href="#">Move from Windows Phone Silverlight to UWP</a> <a href="#">aka.ms/uwpgdg_move-from-windows-phone-silverlight-to-uwp</a>
Porting from XAML or Silverlight to a Universal Windows Platform app (video)	<a href="#">Porting an App from XAML or Silverlight to Windows 10</a> <a href="#">aka.ms/uwpgdg_porting-an-app-from-xaml-or-silverlight</a>
Porting an Xbox game to a Universal Windows Platform app	<a href="#">Porting from Xbox One to Windows 10 UWP</a> <a href="#">aka.ms/uwpgdg_porting-from-xbox-one-to-windows-10-uwp</a>
Porting from DirectX 9 to DirectX 11	<a href="#">Port from DirectX 9 to Universal Windows Platform (UWP)</a> <a href="#">aka.ms/uwpgdg_port-from-directx-to-uwp</a>
Porting from Direct3D 11 to Direct3D 12	<a href="#">Porting from Direct3D 11 to Direct3D 12</a> <a href="#">aka.ms/uwpgdg_port-from-direct3d-11-to-direct3d-12</a>
Porting from OpenGL ES to Direct3D 11	<a href="#">Port from OpenGL ES 2.0 to Direct3D 11</a> <a href="#">aka.ms/uwpgdg_port-from-opengl-es-2.0</a>
OpenGL ES to Direct3D 11 using ANGLE	<a href="#">ANGLE</a> <a href="#">aka.ms/uwpgdg_angle</a>
Classic Windows API equivalents in the UWP	<a href="#">Alternatives to Windows APIs in Universal Windows Platform (UWP) apps</a> <a href="#">aka.ms/uwpgdg_alternatives-to-windows-apis-in-uwp-apps</a>



# Prototype and design

Now that you've decided the type of game you want to create and the tools and graphics technology you'll use to build it, you're ready to get started with the design and prototype. At its core, your game is a Universal Windows Platform app, so that's where you'll begin.

## Introduction to the Universal Windows Platform (UWP)

Windows 10 introduces the Universal Windows Platform (UWP), which provides a common API platform across Windows 10 devices. UWP evolves and expands the Windows Runtime model and hones it into a cohesive, unified core. Games that target the UWP can call WinRT APIs that are common to all devices. Because the UWP provides a guaranteed core API layer, you can choose to create a single app package that will install across Windows 10 devices. And if you want to, your game can still call APIs (including some classic Windows APIs from Win32 and .NET) that are specific to the devices your game runs on.

### The goal of the UWP is to have:

- One core operating system
- One application platform
- One gaming social network
- One store
- One ingestion path

The following are excellent guides that discuss the Universal Windows Platform apps in detail, and are recommended reading to help you understand the platform.

---

Introduction to Universal Windows Platform apps

[What's a Universal Windows Platform app?](#)

[aka.ms/uwpgdg\\_whats-a-universal-windows-platform-app](http://aka.ms/uwpgdg_whats-a-universal-windows-platform-app)

---

Overview of the UWP

[Guide to UWP apps](#)

[aka.ms/uwpgdg\\_guide-to-uwp-apps](http://aka.ms/uwpgdg_guide-to-uwp-apps)

---

## Getting started with UWP development

Getting set up and ready to develop a Universal Windows Platform app is quick and easy. The following guides take you through the process step-by-step.

---

Getting started with UWP development	<a href="#">Get started with Windows apps</a> <a href="https://aka.ms/uwpgdg_get-started-windows-apps">aka.ms/uwpgdg_get-started-windows-apps</a>
Getting set up for UWP development	<a href="#">Get set up</a> <a href="https://aka.ms/uwpgdg_get-set-up">aka.ms/uwpgdg_get-set-up</a>

---

If you're an "absolute beginner" to UWP programming, and are considering using XAML in your game (see [Choosing your graphics technology and programming language \(aka.ms/uwpgdg\\_choosing-your-graphics\)](#)), the [Windows 10 development for absolute beginners \(aka.ms/uwpgdg\\_win10-dev-beginners\)](#) video series is a good place to start.

---

Beginners guide to Windows 10 development with XAML (Video series)	<a href="#">Windows 10 development for absolute beginners</a> <a href="https://aka.ms/uwpgdg_win10-dev-beginners-vid">aka.ms/uwpgdg_win10-dev-beginners-vid</a>
Announcing the Windows 10 absolute beginners series using XAML (blog post)	<a href="#">Windows 10 development for absolute beginners</a> <a href="https://aka.ms/uwpgdg_win10-dev-beginners-blog">aka.ms/uwpgdg_win10-dev-beginners-blog</a>

---

## UWP development concepts

---

Overview of Universal Windows Platform app development	<a href="#">Develop Windows apps</a> <a href="https://aka.ms/uwpgdg_develop-windows-apps">aka.ms/uwpgdg_develop-windows-apps</a>
Overview of network programming in the UWP	<a href="#">Networking and web services</a> <a href="https://aka.ms/uwpgdg_networking-and-web-services">aka.ms/uwpgdg_networking-and-web-services</a>
Using Windows.Web.HTTP and Windows.Networking.Sockets in games	<a href="#">Networking for games</a> <a href="https://aka.ms/Use-Win-WebHTTP-Win-NetworkingSockets-games">aka.ms/Use-Win-WebHTTP-Win-NetworkingSockets-games</a>
Asynchronous programming concepts in the UWP	<a href="#">Asynchronous programming</a> <a href="https://aka.ms/uwpgdg_asynchronous-programming">aka.ms/uwpgdg_asynchronous-programming</a>

---

## Windows Desktop APIs to UWP

These are some links to help you move your Windows desktop game to UWP.

---

Use existing C++ code for UWP game development	<a href="#">How to: Use existing C++ code in a UWP app</a> <a href="#">aka.ms/Use-existing-Cplusplus-UWP-game-dev</a>
UWP APIs for Win32 and COM APIs	<a href="#">Win32 and COM APIs for UWP apps</a> <a href="#">aka.ms/uwpgdg_win32-com-apis</a>
Unsupported CRT functions in UWP	<a href="#">CRT functions not supported in Universal Windows Platform apps</a> <a href="#">aka.ms/uwpgdg_crt-functions-not-supported-uwp</a>
Alternatives for Windows APIs	<a href="#">Alternatives to Windows APIs in Universal Windows Platform (UWP) apps</a> <a href="#">aka.ms/uwpgdg_alternatives-windows-apis-uwp</a>

---

## Process lifetime management

Process lifetime management, or app lifecycle, describes the various activation states that a Universal Windows Platform app can transition through. Your game can be activated, suspended, resumed, or terminated, and can transition through those states in a variety of ways.

---

Handling app lifecycle transitions	<a href="#">App lifecycle</a> <a href="#">aka.ms/uwpgdg_app-lifecycle</a>
Using Microsoft Visual Studio to trigger app transitions	<a href="#">How to trigger suspend, resume, and background events for Windows Store apps in Visual Studio</a> <a href="#">aka.ms/uwpgdg_how-to-trigger-suspend-resume</a>

---

## Designing game UX

The genesis of a great game is inspired design.

Games share some common user interface elements and design principles with apps, but games often have a unique look, feel, and design goal for their user experience. Games succeed when thoughtful design is applied to both aspects—when should your game use tested UX, and when should it diverge and innovate? The presentation technology that you choose for your game—DirectX, XAML, HTML5, or some combination of the three—will influence implementation details, but the design principles you apply are largely independent of that choice.

Separately from UX design, gameplay design such as level design, pacing, world design, and other aspects is an art form of its own—one that's up to you and your team, and not covered in this development guide.

---

UWP design basics and guidelines	<a href="#">Designing UWP apps</a> <a href="https://aka.ms/uwpgdg_designing-uwp-apps">aka.ms/uwpgdg_designing-uwp-apps</a>
Designing for app lifecycle states	<a href="#">UX guidelines for launch, suspend, and resume</a> <a href="https://aka.ms/uwpgdg_ux-guidelines-for-launch-suspend-resume">aka.ms/uwpgdg_ux-guidelines-for-launch-suspend-resume</a>
Design your UWP app for Xbox One and television screens	<a href="#">Designing for Xbox and TV</a> <a href="https://aka.ms/Design-UWP-app-xbox1-tvscreens">aka.ms/Design-UWP-app-xbox1-tvscreens</a>
Targeting multiple device form factors (video)	<a href="#">Designing Games for a Windows Core World</a> <a href="https://aka.ms/uwpgdg_designing-games-for-a-wcw">aka.ms/uwpgdg_designing-games-for-a-wcw</a>

---

### Color guideline and palette

Following a consistent color guideline in your game improves aesthetics, aids navigation, and is a powerful tool to inform the player of menu and HUD functionality. Consistent coloring of game elements like warnings, damage, XP, and achievements can lead to cleaner UI and reduce the need for explicit labels.

---

Color guide	<a href="#">Best Practices: Color</a> <a href="https://aka.ms/uwpgdg_best-practices-color">aka.ms/uwpgdg_best-practices-color</a>
-------------	--

---



## Typography

The appropriate use of typography enhances many aspects of your game, including UI layout, navigation, readability, atmosphere, brand, and player immersion.

---

Typography guide

[Best Practices: Typography](#)

[aka.ms/uwpgdg\\_best-practices-typography](https://aka.ms/uwpgdg_best-practices-typography)

---

## UI map

A UI map is a layout of game navigation and menus expressed as a flowchart. The UI map helps all involved stakeholders understand the game's interface and navigation paths, and can expose potential roadblocks and dead ends early in the development cycle.

---

UI map guide

[Best Practices: UI Map](#)

[aka.ms/uwpgdg\\_best-practices-ui-map](https://aka.ms/uwpgdg_best-practices-ui-map)

---

## Game Audio

Guides and references for implementing audio in games using XAudio2, XAPO, and Windows Sonic. XAudio2 is a low-level audio API that provides signal processing and mixing foundation for developing high performance audio engines. XAPO API allows the creation of cross-platform audio processing objects (XAPO) for use in XAudio2 on both Windows and Xbox. Windows Sonic audio support allows you to add Dolby Atmos for Home Theater, Dolby Atmos for Headphones, and Windows HRTF support to your game or streaming media application.

---

XAudio2 APIs

[Programming guide and API reference for XAudio2](#)

[aka.ms/XAudio2-APIs](https://aka.ms/XAudio2-APIs)

---

Create cross-platform audio processing objects

[XAPO Overview](#)

[aka.ms/Cre-at-crossplatform-audio-processing-objects](https://aka.ms/Cre-at-crossplatform-audio-processing-objects)

---

Intro to audio concepts

[Audio for games](#)

[aka.ms/Intro-audio-concepts](https://aka.ms/Intro-audio-concepts)

---

Windows Sonic overview	<a href="#">Spatial sound</a> <a href="#">aka.ms/Windows-sonic-overview</a>
Windows Sonic spatial sound samples	<a href="#">Xbox Advanced Technology Group audio samples</a> <a href="#">aka.ms/Windows-sonic-spatial-sound-samples</a>
Learn how to integrate Windows Sonic into your games (video)	<a href="#">Introducing Spatial Audio Capabilities for Xbox and Windows</a> <a href="#">aka.ms/Learn-to-integrate-WinSonic-into-games</a>

## DirectX development

Guides and references for DirectX game development.

DirectX game development on the UWP	<a href="#">Games and DirectX</a> <a href="#">aka.ms/uwpgdg_games-and-directx</a>
DirectX interaction with the UWP app model	<a href="#">The app object and DirectX</a> <a href="#">aka.ms/uwpgdg_app-object-and-directx</a>
Graphics and DirectX 12 development videos (YouTube channel)	<a href="#">Microsoft DirectX 12 and Graphics Education</a> <a href="#">aka.ms/uwpgdg_microsoft-directx-12-and-graphics-education</a>
DirectX overviews and reference	<a href="#">DirectX Graphics and Gaming</a> <a href="#">aka.ms/uwpgdg_directx-graphics-and-gaming</a>
Direct3D 12 programming guide and reference	<a href="#">Direct3D 12 Graphics</a> <a href="#">aka.ms/uwpgdg_direct3d-12-graphics</a>
DirectX 12 fundamentals (video)	<a href="#">Better Power, Better Performance: Your Game on DirectX 12</a> <a href="#">aka.ms/uwpgdg_better-power-better-performance</a>

## Learning Direct3D 12

Learn what changed in Direct3D 12 and how to start programming using Direct3D 12.

Set up programming environment	<a href="#">Direct3D 12 programming environment setup</a> <a href="#">aka.ms/uwpgdg_direct3d-12-programming-environment-setup</a>
How to create a basic component	<a href="#">Creating a basic Direct3D 12 component</a> <a href="#">aka.ms/uwpgdg_creating-a-basic-direct3d-12-component</a>
Changes in Direct3D 12	<a href="#">Important changes migrating from Direct3D 11 to Direct3D 12</a> <a href="#">aka.ms/uwpgdg_important-changes-migrating</a>
How to port from Direct3D 11 to Direct3D 12	<a href="#">Porting from Direct3D 11 to Direct3D 12</a> <a href="#">aka.ms/uwpgdg_porting-from-direct3d-11</a>
Resource-binding concepts (covering descriptor, descriptor table, descriptor heap, and root signature)	<a href="#">Resource binding in Direct3D 12</a> <a href="#">aka.ms/uwpgdg_resource-binding-in-direct3d-12</a>
Managing memory	<a href="#">Memory management in Direct3D 12</a> <a href="#">aka.ms/Managing-memory</a>

## DirectX Tool Kit and libraries

The DirectX Tool Kit, DirectX texture-processing library, DirectXMesh geometry-processing library, UVAAtlas library, and DirectXMath library provide texture, mesh, sprite, and other utility functionality and helper classes for DirectX development. These libraries can help you save development time and effort.

Get DirectX Tool Kit for DirectX 11	<a href="#">DirectXTK</a> <a href="#">aka.ms/uwpgdg_directxtk</a>
Get DirectX Tool Kit for DirectX 12	<a href="#">DirectXTK 12</a> <a href="#">aka.ms/uwpgdg_directxtk-12</a>
Get DirectX texture-processing library	<a href="#">DirectXTex</a> <a href="#">aka.ms/uwpgdg_directxtex</a>
Get DirectXMesh geometry-processing library	<a href="#">DirectXMesh</a> <a href="#">aka.ms/uwpgdg_directxmesh</a>

---

Get UVAtlas for creating and packing isochart texture atlas	<a href="#">UVAtlas</a> <a href="https://aka.ms/uwpgdg_uvAtlas">aka.ms/uwpgdg_uvAtlas</a>
Get the DirectXMath library	<a href="#">DirectXMath</a> <a href="https://aka.ms/uwpgdg_directxmath">aka.ms/uwpgdg_directxmath</a>
Direct3D 12 support in the DirectXTK (blog post)	<a href="#">Support for DirectX 12</a> <a href="https://aka.ms/uwpgdg_support-for-directx-12">aka.ms/uwpgdg_support-for-directx-12</a>

---

## DirectX resources from partners

These are some additional DirectX documentation created by external partners.

---

Nvidia: DX12 Do's and Don'ts (blog post)	<a href="#">DirectX 12 on Nvidia GPUs</a> <a href="https://aka.ms/uwpgdg_directx-12-on-nvidia-gpus">aka.ms/uwpgdg_directx-12-on-nvidia-gpus</a>
Intel: Efficient rendering with DirectX 12	<a href="#">DirectX 12 rendering on Intel Graphics</a> <a href="https://aka.ms/uwpgdg_directx-12-rendering-on-intel-graphics">aka.ms/uwpgdg_directx-12-rendering-on-intel-graphics</a>
Intel: Multi-adapter support in DirectX 12	<a href="#">How to implement an explicit multi-adapter application using DirectX 12</a> <a href="https://aka.ms/uwpgdg_how-to-implement-explicit-multi-adapter">aka.ms/uwpgdg_how-to-implement-explicit-multi-adapter</a>
Intel: DirectX 12 tutorial	<a href="#">Collaborative white paper by Intel, Suzhou Snail, and Microsoft</a> <a href="https://aka.ms/uwpgdg_collaborative-white-paper">aka.ms/uwpgdg_collaborative-white-paper</a>

---



# Production

Your studio is now fully engaged and moving into the production cycle, with work distributed throughout your team. You're polishing, refactoring, and extending the prototype to craft it into a full game.

## Notifications and live tiles

A tile is your game's representation on the Start Menu. Tiles and notifications can drive player interest even when they aren't currently playing your game.

Developing tiles and badges	<a href="https://aka.ms/uwpgdg_tiles-badges-notifications">Tiles, badges, and notifications</a> <a href="https://aka.ms/uwpgdg_tiles-badges-notifications">aka.ms/uwpgdg_tiles-badges-notifications</a>
Sample illustrating live tiles and notifications	<a href="https://aka.ms/uwpgdg_notifications-sample">Notifications sample</a> <a href="https://aka.ms/uwpgdg_notifications-sample">aka.ms/uwpgdg_notifications-sample</a>
Adaptive tile templates (blog post)	<a href="https://aka.ms/uwpgdg_adaptive-tile-templates">Adaptive Tile Templates—Schema and Documentation</a> <a href="https://aka.ms/uwpgdg_adaptive-tile-templates">aka.ms/uwpgdg_adaptive-tile-templates</a>
Designing tiles and badges	<a href="https://aka.ms/uwpgdg_guidelines-for-tiles-and-badges">Guidelines for tiles and badges</a> <a href="https://aka.ms/uwpgdg_guidelines-for-tiles-and-badges">aka.ms/uwpgdg_guidelines-for-tiles-and-badges</a>
Windows 10 app for interactively developing live tile templates	<a href="https://aka.ms/uwpgdg_notifications-visualizer">Notifications Visualizer</a> <a href="https://aka.ms/uwpgdg_notifications-visualizer">aka.ms/uwpgdg_notifications-visualizer</a>
UWP Tile Generator extension for Visual Studio	<a href="https://aka.ms/uwpgdg_tool-for-creating-all-required-tiles">Tool for creating all required tiles using single image</a> <a href="https://aka.ms/uwpgdg_tool-for-creating-all-required-tiles">aka.ms/uwpgdg_tool-for-creating-all-required-tiles</a>
UWP Tile Generator extension for Visual Studio (blog post)	<a href="https://aka.ms/uwpgdg_tips-on-using-uwp-tile-generator-tool">Tips on using the UWP Tile Generator tool</a> <a href="https://aka.ms/uwpgdg_tips-on-using-uwp-tile-generator-tool">aka.ms/uwpgdg_tips-on-using-uwp-tile-generator-tool</a>

## Enable in-app product (IAP) purchases

An IAP (in-app product) is a supplementary item that players can purchase in-game. IAPs can be new add-ons, game levels, items, or anything else that your players might enjoy. Used appropriately, IAPs can provide revenue while improving the game experience. You define and publish your game's IAPs through the Windows Dev Center dashboard, and enable in-app purchases in your game's code.

---

Durable in-app products	<a href="#">Enable in-app product purchases</a> <a href="https://aka.ms/uwpgdg_enable-in-app-product-purchases">aka.ms/uwpgdg_enable-in-app-product-purchases</a>
Consumable in-app products	<a href="#">Enable consumable in-app product purchases</a> <a href="https://aka.ms/uwpgdg_enable-consumable-in-app-product-purchases">aka.ms/uwpgdg_enable-consumable-in-app-product-purchases</a>
In-app product details and submission	<a href="#">IAP submissions</a> <a href="https://aka.ms/uwpgdg_iap-submissions">aka.ms/uwpgdg_iap-submissions</a>
Monitor IAP sales and demographics for your game	<a href="#">IAP acquisitions report</a> <a href="https://aka.ms/uwpgdg_iap-acquisitions-report">aka.ms/uwpgdg_iap-acquisitions-report</a>

---

## Debugging, performance optimization, and monitoring

To optimize performance, take advantage of Game Mode in Windows 10 to provide your gamers with the best possible gaming experience by fully utilizing the capacity of their current hardware.

The Windows Performance Toolkit (WPT) consists of performance monitoring tools that produce in-depth performance profiles of Windows operating systems and applications. This is especially useful for monitoring memory usage and improving game performance. The Windows Performance Toolkit is included in the Windows 10 SDK and Windows ADK. This toolkit consists of two independent tools: Windows Performance Recorder (WPR) and Windows Performance Analyzer (WPA). ProcDump, which is part of [Windows Sysinternals](#) ([aka.ms/uwpgdg\\_windows-sysinternals](https://aka.ms/uwpgdg_windows-sysinternals)), is a command-line utility that monitors CPU spikes and generates dump files during game crashes.

---

Performance test your code	<a href="#">Cloud based load testing</a> <a href="https://aka.ms/Performance-test-code">aka.ms/Performance-test-code</a>
Improve performance by getting exclusive or priority access to hardware resources using Game Mode APIs	<a href="#">Game Mode</a> <a href="https://aka.ms/Improve-performance-access-hardware-resources-GameModeAPIs">aka.ms/Improve-performance-access-hardware-resources-GameModeAPIs</a>

---

Get Windows Performance Toolkit (WPT) from Windows 10 SDK	<a href="#">Windows 10 SDK</a> <a href="https://aka.ms/uwpgdg_windows-10-sdk">aka.ms/uwpgdg_windows-10-sdk</a>
Get Windows Performance Toolkit (WPT) from Windows ADK	<a href="#">Windows ADK</a> <a href="https://aka.ms/uwpgdg_windows-adk">aka.ms/uwpgdg_windows-adk</a>
Troubleshoot unresponsive UI using Windows Performance Analyzer (video)	<a href="#">Critical path analysis with WPA</a> <a href="https://aka.ms/uwpgdg_critical-path-analysis-with-wpa">aka.ms/uwpgdg_critical-path-analysis-with-wpa</a>
Diagnose memory usage and leaks using Windows Performance Recorder (video)	<a href="#">Memory footprint and leaks</a> <a href="https://aka.ms/uwpgdg_memory-footprint-and-leaks">aka.ms/uwpgdg_memory-footprint-and-leaks</a>
Get ProcDump	<a href="#">ProcDump</a> <a href="https://aka.ms/uwpgdg_procdump">aka.ms/uwpgdg_procdump</a>
Learn to use ProcDump (video)	<a href="#">Configure ProcDump to create dump files</a> <a href="https://aka.ms/uwpgdg_configure-procdump-to-create-dump-files">aka.ms/uwpgdg_configure-procdump-to-create-dump-files</a>

## Advanced DirectX techniques and concepts

Some portions of DirectX development can be nuanced and complex. When you get to the point in production where you need to dig down into the details of your DirectX engine, or debug difficult performance problems, the resources and information in this section can help.

PIX on Windows	<a href="#">Performance tuning and debugging tool for DirectX 12 on Windows</a> <a href="https://aka.ms/PIX-on-Windows">aka.ms/PIX-on-Windows</a>
Debugging and validation tools for D3D12 development (video)	<a href="#">D3D12 Performance Tuning and Debugging with PIX and GPU Validation</a> <a href="https://aka.ms/Debugging-validation-tools-D3D12-Dev">aka.ms/Debugging-validation-tools-D3D12-Dev</a>
Optimizing graphics and performance (video)	<a href="#">Advanced DirectX 12 Graphics and Performance</a> <a href="https://aka.ms/uwpgdg_advanced-directx-12-graphics-performance">aka.ms/uwpgdg_advanced-directx-12-graphics-performance</a>
DirectX graphics debugging (video)	<a href="#">Solve the tough graphics problems with your game using DirectX Tools</a> <a href="https://aka.ms/uwpgdg_solve-the-tough-graphics-problem">aka.ms/uwpgdg_solve-the-tough-graphics-problem</a>

---

Visual Studio 2015 tools for debugging DirectX 12 (video)

[DirectX tools for Windows 10 in Visual Studio 2015](https://aka.ms/uwpgdg_directx-tools-for-windows-10-visual-studio)  
[aka.ms/uwpgdg\\_directx-tools-for-windows-10-visual-studio](https://aka.ms/uwpgdg_directx-tools-for-windows-10-visual-studio)

---

Direct3D 12 programming guide

[Direct3D 12 Programming Guide](https://aka.ms/uwpgdg_direct3d-12-programming-guide)  
[aka.ms/uwpgdg\\_direct3d-12-programming-guide](https://aka.ms/uwpgdg_direct3d-12-programming-guide)

---

Combining DirectX and XAML

[DirectX and XAML interop](https://aka.ms/uwpgdg_directx-and-xaml-interop)  
[aka.ms/uwpgdg\\_directx-and-xaml-interop](https://aka.ms/uwpgdg_directx-and-xaml-interop)

---

## Globalization and localization

Develop world-ready games for the Windows platform and learn about the international features built into Microsoft's top products.

---

Preparing your game for the global market

[Guidelines when developing for a global audience](https://aka.ms/uwpgdg_guidelines-when-developing-global-audience)  
[aka.ms/uwpgdg\\_guidelines-when-developing-global-audience](https://aka.ms/uwpgdg_guidelines-when-developing-global-audience)

---

Bridging languages, cultures, and technology

[Online resource for language conventions and standard Microsoft terminology](https://aka.ms/uwpgdg_online-resource-for-language-conventions)  
[aka.ms/uwpgdg\\_online-resource-for-language-conventions](https://aka.ms/uwpgdg_online-resource-for-language-conventions)

---





# Submitting and publishing your game

The following guides and information help make the publishing and submission process as smooth as possible.

## Publishing

You'll use the new unified Windows Dev Center dashboard to publish and manage your game packages.

Windows Dev Center app publishing	<a href="#">Publish Windows apps</a> <a href="https://aka.ms/uwpgdg_publish-windows-apps">aka.ms/uwpgdg_publish-windows-apps</a>
Windows Dev Center advanced publishing (GDN)	<a href="#">Windows Dev Center Dashboard advanced publishing guide</a> <a href="https://aka.ms/uwpgdg_windows-dev-center-dashboard">aka.ms/uwpgdg_windows-dev-center-dashboard</a>
Use Azure Active Directory (AAD) to add users to your Dev Center account	<a href="#">Manage account users</a> <a href="https://aka.ms/AAD-add-users-DevCenter-account">aka.ms/AAD-add-users-DevCenter-account</a>
Rating your game (blog post)	<a href="#">Single workflow to assign age ratings using IARC system</a> <a href="https://aka.ms/uwpgdg_single-workflow-to-assign-age-ratings">aka.ms/uwpgdg_single-workflow-to-assign-age-ratings</a>

## Packaging and uploading

Learn to use streaming install and optional packages (video)	<a href="#">Nextgen UWP app distribution: Building extensible, stream-able, componentized apps</a> <a href="https://aka.ms/Learn-streaming-install-optional-packages">aka.ms/Learn-streaming-install-optional-packages</a>
Divide and group content to enable streaming install	<a href="#">UWP App Streaming install</a> <a href="https://aka.ms/Divide-group-content-enable-streaming-install">aka.ms/Divide-group-content-enable-streaming-install</a>
Create optional packages like DLC game content	<a href="#">Optional packages and related set authoring</a> <a href="https://aka.ms/Create-optional-packages-DLC-game-content">aka.ms/Create-optional-packages-DLC-game-content</a>
Package your UWP game	<a href="#">Packaging apps</a> <a href="https://aka.ms/Package-UWP-game">aka.ms/Package-UWP-game</a>

---

Package your UWP DirectX game

[Package your UWP DirectX game](#)  
[aka.ms/Package-UWP-Directx-game](#)

---

Packaging your game as a 3rd party developer  
(blog post)

[Create uploadable packages without publisher's store account access](#)  
[aka.ms/uwpgdg\\_create-uploadable-packages-without](#)

---

Creating app packages and app package bundles using MakeAppx

[Create packages using app packager tool MakeAppx.exe](#)  
[aka.ms/Create-app-package-bundles-MakeAppx](#)

---

Signing your files digitally using SignTool

[Sign files and verify signatures in files using SignTool](#)  
[aka.ms/uwpgdg\\_sign-files-and-verify-signatures](#)

---

Uploading and versioning your game

[Upload app packages](#)  
[aka.ms/uwpgdg\\_upload-app-packages](#)

---

## Policies and certification

Don't let certification issues delay your game's release. Here are policies and common certification issues to be aware of.

---

Windows Store App Developer Agreement

[App Developer Agreement](#)  
[aka.ms/uwpgdg\\_app-developer-agreement](#)

---

Policies for publishing apps in the Windows Store

[Windows Store Policies](#)  
[aka.ms/Windows-store-policies](#)

---

How to avoid some common app certification issues

[Avoid common certification failures](#)  
[aka.ms/uwpgdg\\_avoid-common-certification-failures](#)

---

## Store manifest (StoreManifest.xml)

The store manifest (StoreManifest.xml) is an optional configuration file that can be included in your app package. The store manifest provides additional features that are not part of the AppxManifest.xml file. For example, you can use the store manifest to block installation of your game if a target device doesn't have the specified minimum DirectX feature level, or the specified minimum system memory.

---

Store manifest schema

[StoreManifest schema \(Windows 10\)](#)  
[aka.ms/uwpgdg\\_storemanifest-schema](#)



# Game lifecycle management

After you've finished development and shipped your game, it's not "game over." You may be done with development on version one, but your game's journey in the marketplace has only just begun. You'll want to monitor usage and error reporting, respond to user feedback, and publish updates to your game.

## Windows Dev Center analytics and promotion

---

Dev Center App	<a href="#">Dev Center Windows 10 app to view performance of your published apps</a> <a href="#">aka.ms/uwpgdg_dev-center-windows-10-app-to-view</a>
Windows Dev Center analytics	<a href="#">Analytics</a> <a href="#">aka.ms/uwpgdg_analytics</a>
Responding to customer reviews	<a href="#">Respond to customer reviews</a> <a href="#">aka.ms/uwpgdg_respond-to-customer-reviews</a>
Ways to promote your game	<a href="#">Promote your apps</a> <a href="#">aka.ms/uwpgdg_promote-your-apps</a>

---

## Visual Studio Application Insights

Visual Studio Application Insights provides performance, telemetry, and usage analytics for your published game. Application Insights helps you detect and solve issues after your game is released, continuously monitor and improve usage, and understand how players are continuing to interact with your game. Application Insights works by adding an SDK into your app, which sends telemetry to the [Azure portal](#) ([aka.ms/uwpgdg\\_azure-portal](#)).

---

Application performance and usage analytics	<a href="#">Visual Studio Application Insights</a> <a href="#">aka.ms/uwpgdg_visual-studio-application-insights</a>
Enable Application Insights in Windows apps	<a href="#">Application Insights for Windows Phone and Store apps</a> <a href="#">aka.ms/uwpgdg_application-insights-windows-phone-store</a>

---

## Third party solutions for analytics and promotion

---

Understand player behavior using GameAnalytics	<a href="#">GameAnalytics</a> <a href="https://aka.ms/Understand-player-behavior-GameAnalytics">aka.ms/Understand-player-behavior-GameAnalytics</a>
Connect your UWP game to Google Analytics	<a href="#">Get Windows SDK for Google Analytics</a> <a href="https://aka.ms/Connect-UWP-game-Google-Analytics">aka.ms/Connect-UWP-game-Google-Analytics</a>
Learn how to use Windows SDK for Google Analytics (video)	<a href="#">Getting started with Windows SDK for Google Analytics</a> <a href="https://aka.ms/Learn-WindowsSDK-GoogleAnalytics">aka.ms/Learn-WindowsSDK-GoogleAnalytics</a>
Use Facebook App Installs Ads to promote your game to Facebook users	<a href="#">Get Windows SDK for Facebook</a> <a href="https://aka.ms/Facebook-install-Ads-promote-game">aka.ms/Facebook-install-Ads-promote-game</a>
Learn how to use Facebook App Installs Ads (video)	<a href="#">Getting started with Windows SDK for Facebook</a> <a href="https://aka.ms/Learn-Facebook-install-Ads">aka.ms/Learn-Facebook-install-Ads</a>
Use Vungle to add video ads into your games	<a href="#">Get Windows SDK for Vungle</a> <a href="https://aka.ms/Vungle-video-adds-game">aka.ms/Vungle-video-adds-game</a>

---

## Creating and managing content updates

To update your published game, submit a new app package with a higher version number. After the package makes its way through submission and certification, it will automatically be available to customers as an update.

---

Updating and versioning your game	<a href="#">Package version numbering</a> <a href="https://aka.ms/package-version-numbering">aka.ms/package-version-numbering</a>
Game package management guidance	<a href="#">Guidance for app package management</a> <a href="https://aka.ms/guidance-app-package-management">aka.ms/guidance-app-package-management</a>

---



# Adding Xbox Live to your game

Xbox Live is a premier gaming network that connects millions of gamers across the world. Developers gain access to Xbox Live features that can organically grow their game's audience, including Xbox Live presence, Leaderboards, Cloud Saves, Game Hubs, Clubs, Party Chat, Game DVR, and more.

## NOTE

If you would like to develop Xbox Live enabled titles, several options are available to you. For info about the various programs, see [Developer program overview \(aka.ms/Developer-Program-Overview\)](https://aka.ms/Developer-Program-Overview).

<a href="#">Xbox Live overview</a>	<a href="#">Xbox Live developer guide</a> <a href="https://aka.ms/Xboxlive-overview">aka.ms/Xboxlive-overview</a>
<a href="#">Understand which features are available depending on program</a>	<a href="#">Developer program overview: Feature table</a> <a href="https://aka.ms/Understand-available-features-depending-on-program">aka.ms/Understand-available-features-depending-on-program</a>
<a href="#">Links to useful resources for developing Xbox Live games</a>	<a href="#">Xbox Live Resources</a> <a href="https://aka.ms/Useful-resources-dev-XboxLive-games">aka.ms/Useful-resources-dev-XboxLive-games</a>
<a href="#">Learn how to get info from Xbox Live services</a>	<a href="#">Introduction to Xbox Live APIs</a> <a href="https://aka.ms/Learn-to-get-info-Xboxlive-services">aka.ms/Learn-to-get-info-Xboxlive-services</a>

## For developers in the Xbox Live Creators Program

---

Overview	<a href="https://aka.ms/Overview-Xboxlive-creators">Get started with the Xbox Live Creators Program</a> <a href="https://aka.ms/Overview-Xboxlive-creators">aka.ms/Overview-Xboxlive-creators</a>
Add Xbox Live to your game	<a href="https://aka.ms/Add-Xboxlive-your-game">Step by step guide to integrate Xbox Live Creators program</a> <a href="https://aka.ms/Add-Xboxlive-your-game">aka.ms/Add-Xboxlive-your-game</a>
Add Xbox Live to your UWP game created using Unity	<a href="https://aka.ms/Xboxlive-UWPgame-Unity">Get started developing an Xbox Live Creators Program title with the Unity game engine</a> <a href="https://aka.ms/Xboxlive-UWPgame-Unity">aka.ms/Xboxlive-UWPgame-Unity</a>
Set up your development sandbox	<a href="https://aka.ms/Dev-Sandbox-setup">Xbox live sandboxes introduction</a> <a href="https://aka.ms/Dev-Sandbox-setup">aka.ms/Dev-Sandbox-setup</a>
Set up accounts for testing	<a href="https://aka.ms/Setup-accounts-testing">Authorize Xbox Live accounts in your test environment</a> <a href="https://aka.ms/Setup-accounts-testing">aka.ms/Setup-accounts-testing</a>
Samples for Xbox Live Creators Program	<a href="https://aka.ms/Samples-XboxLive-Creators-Program">Code samples for Creators Program developers</a> <a href="https://aka.ms/Samples-XboxLive-Creators-Program">aka.ms/Samples-XboxLive-Creators-Program</a>
Learn how to integrate cross-platform Xbox Live experiences in UWP games (video)	<a href="https://aka.ms/Integrate-crossplatform-Xboxlive-UWPgames">Xbox Live Creators Program</a> <a href="https://aka.ms/Integrate-crossplatform-Xboxlive-UWPgames">aka.ms/Integrate-crossplatform-Xboxlive-UWPgames</a>

---

## For managed partners and developers in the ID@Xbox program

---

Overview	<a href="https://aka.ms/Overview-Xboxlive-partner">Get started with Xbox Live as a managed partner or an ID developer</a> <a href="https://aka.ms/Overview-Xboxlive-partner">aka.ms/Overview-Xboxlive-partner</a>
Add Xbox Live to your game	<a href="https://aka.ms/Xboxlive-your-game">Step by step guide to integrate Xbox Live for managed partners and ID members</a> <a href="https://aka.ms/Xboxlive-your-game">aka.ms/Xboxlive-your-game</a>

---

---

Add Xbox Live to your UWP game created using Unity	<a href="#">Add Xbox Live support to Unity for UWP with IL2CPP scripting backend for ID and managed partners</a> <a href="#">aka.ms/Xboxlive-UWPgame-created-Unity</a>
Set up your development sandbox	<a href="#">Advanced Xbox Live sandboxes</a> <a href="#">aka.ms/Setup-Dev-Sandbox</a>
Requirements for games that use Xbox Live (GDN)	<a href="#">Xbox Requirements for Xbox Live on Windows 10</a> <a href="#">aka.ms/game-requirements-Xboxlive-GDN</a>
Samples	<a href="#">Code samples for ID@Xbox developers</a> <a href="#">aka.ms/XboxLive-samples2</a>
Overview of Xbox Live game development (video)	<a href="#">Developing with Xbox Live for Windows 10</a> <a href="#">aka.ms/uwpgdg_developing-with-xbox-live-for-windows10</a>
Cross-platform matchmaking (video)	<a href="#">Xbox Live Multiplayer: Introducing services for cross-platform matchmaking and gameplay</a> <a href="#">aka.ms/uwpgdg_xbox-live-multiplayer-introducing</a>
Cross-device gameplay in Fable Legends (video)	<a href="#">Fable Legends: Cross-device Gameplay with Xbox Live</a> <a href="#">aka.ms/CrossDevice-gameplay-FableLegends</a>
Xbox Live stats and achievements (video)	<a href="#">Best Practices for Leveraging Cloud-Based User Stats and Achievements in Xbox Live</a> <a href="#">aka.ms/uwpgdg_best-practices-leveraging-cloud-based-user</a>

---



# Additional resources

Game development videos

Videos from major conferences like GDC and //build  
[aka.ms/game-dev-videos](https://aka.ms/game-dev-videos)

Indie game development (video)

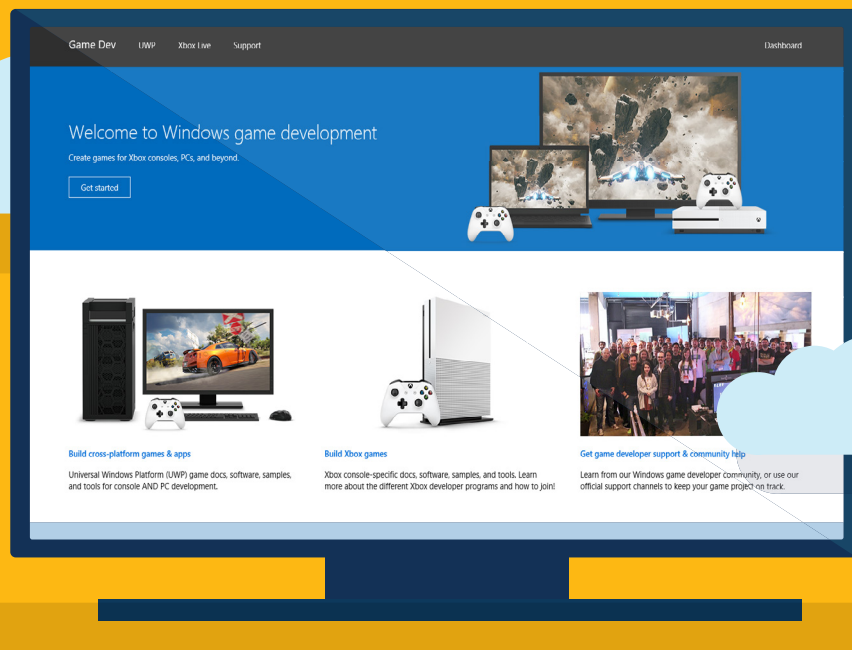
New Opportunities for Independent Developers  
[aka.ms/uwpgdg\\_new-opportunities-for-independent-dev](https://aka.ms/uwpgdg_new-opportunities-for-independent-dev)

Considerations for multi-core mobile devices (video)

Sustained Gaming Performance in multi-core mobile devices  
[aka.ms/uwpgdg\\_sustained-gaming-performance](https://aka.ms/uwpgdg_sustained-gaming-performance)

Developing Windows 10 desktop games (video)

PC Games for Windows 10  
[aka.ms/uwpgdg\\_pc-games-for-windows-10](https://aka.ms/uwpgdg_pc-games-for-windows-10)



[developer.microsoft.com/games](https://developer.microsoft.com/games)





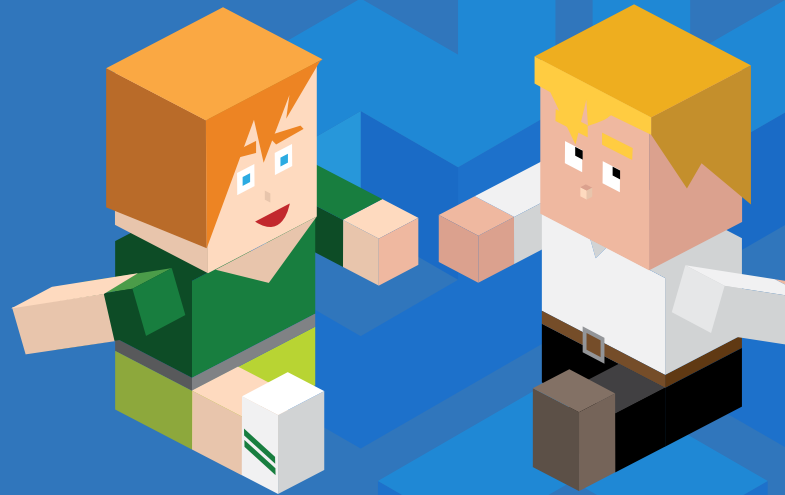
# Dream.Build.Play

## Dream.Build.Play Community

Dream.Build.Play is an online community of creators and game makers from across the globe. It was created under the premise that together, as a global collective of talented individuals, we can elevate the industry with great games and content.

### Dream.Build.Play is a:

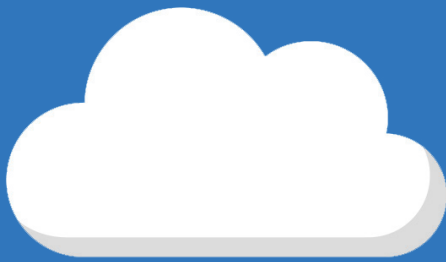
- Global community of game makers
- Resource for tutorials and guides
- Host of contests and challenges reaching more than \$200,000 in prizes
- Place to showcase and share game ideas, content, artwork and more



## Dream.Build.Play 2017 Challenge

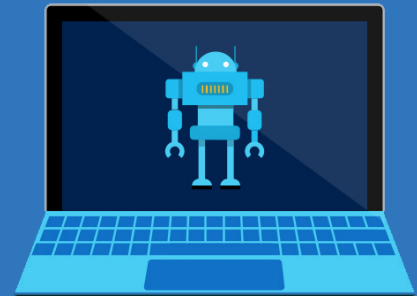
As a Dream.Build.Play community member, you have access to hundreds of thousands in prizes, starting with the launch of the 2017 Challenge. Showcase your game, submit it by December 31, 2017, and compete to win prizes totaling over \$200K.

### Four game category grand prizes:



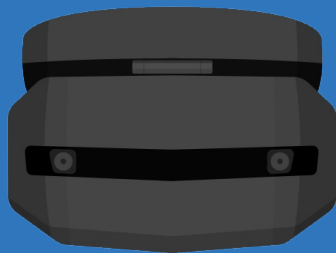
#### Cloud-powered game

Grand prize: \$100,000



#### PC game

Grand prize: \$50,000



#### Mixed reality game

Grand prize: \$50,000



#### Console game

Grand prize: \$25,000

Join our community of game makers: [dreambuildplay.com](http://dreambuildplay.com)



## Microsoft Game Development Guide