

What is Uno Platform?

Uno Platform is the Universal Windows Platform Bridge that allows UWP-based code (C# and XAML) to run on iOS, Android, and WebAssembly.

It provides the full API definitions of UWP Windows 10 2004 (19041) and the implementation of parts of the UWP API, such as Windows.UI.Xaml, to allow UWP applications to run on these platforms.

This allows to use UWP tools from Windows in Visual Studio, such as Xaml Edit and Continue and C# Edit and Continue, build as many apps as possible on Windows, then verify that the app runs on iOS, Android and WebAssembly.

XAML User Interface (UI) provides the ability to display the same XAML files on Windows, iOS, Android and WebAssembly platforms. Uno also provides support for Model-View-ViewModel (MVVM) patterns on all platforms, with features for binding, styling, controlling, and template data generation.

Since the Uno Platform provides all the complete UWP platform APIs, any UWP library can be compiled on Uno (e.g. XamlBehaviors), with the ability to determine which APIs are implemented or not. via IDE using C# Analyzer.



Why use Uno Platform?

Development for Windows (phone, desktop, tablet, Xbox), WebAssembly, iOS (tablet and phone), Android (tablet and phone), macOS, and Linux at the same time can be a complicated process, especially when it comes to user interfaces. Each platform has its own way of defining dynamic layouts that can be effective, elegant, or different in performance.

However, being able to master all of these frameworks at once is a particularly difficult task, as the amount of specific knowledge required to master each platform is immense. In almost all cases, different teams develop the same application over and over, each requiring a full development cycle.

With the 'classic' Xamarin, C# comes to all these platforms, however, it only provides transparent transitions of the UI frameworks available for iOS and Android. Most of the non-UI code can be shared, but when it comes to

the UI, almost nothing can be shared.

In order to avoid having to learn the approach and techniques of creating UI layouts for each platform, Uno.UI mimics the approach of Windows XAML for defining UI and layout. Uno.UI is capable of sharing styles, layouts, and data binding while retaining the ability to combine XAML styles and original layouts. For example, StackPanel can easily contain RelativeLayout on Android or MKMapView on iOS.

Uno.UI provides the ability for developers to reuse known coding and layout techniques across all platforms, increasing overall productivity when creating UI-centric applications. .

What can't Uno Platform do?

Uno is not meant to completely replace all of the original UI frameworks. This would be the lowest denominator approach and result in the end user noticing the non-native app appearance or behavior on their device. There's an iOS app that works like an Android app that can annoy users.

Uno provides a common set of layouts and controls, designed to provide the ability to share an important part of your application's code and markup. However, it still leaves the developers with the ability to keep the original interface. At the same time, Uno provides a way to have a perfectly detailed UI and UX across all platforms. Typically, this theme will be found in the navigation, transitions and animations, the main page, and the edges of the screen.

Although the Uno Platform provides all of the UWP APIs, many of them are not implemented. It currently provides a small set of basic UWP non-UI parts, such as the Windows.UI.Xaml.Application class, which gives the ability to have a generic application bootstrap code.

How does Uno Platform work?



Uno provides a set of APIs that use Windows UWP-compatible class and property names and allow those classes to inherit from the platform's original layout container, in the case of the XAML APIs.

For example, Windows.UI.Xaml.Controls.StackPanel inherits directly from ViewGroup on Android and UIView on iOS.

The original layout system for internal elements is then overwritten with the XAML-compatible layout system. This means a StackPanel will use the same layout strategies on all platforms.

On Windows platforms, Uno.UI is not present and the XAML layout files are preserved. On classic Xamarin-compatible platforms, XAML files are processed at compile-time to generate unconditional code that will be executed at runtime on the device. This means there is no XAML run-time parsing, which makes creating the UI tree particularly efficient.

Uno.UI also provides ways to markup platform-specific XAML files, allowing to create a simple file tree while tailoring the user interface for each platform.

You finished reading the article "**What is Uno Platform?**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.