Windows Presentation

Foundation with .NET

Windows Presentation Foundation with .NET

.NET

.NET is a free, modern open-source cross-platform framework from Microsoft that allows developers to build applications and services for Cloud, Mobile, IoT and more along with Web and Desktop. .NET includes improvements or new features every year with even-numbered long-term releases supported for three years and odd numbered short-term releases supported for eighteen months.

.NET includes C# which is the open-source programming language for modern software development featuring type safety, asynchronous programming and is also updated each year. .NET applications can be developed for any device or platform such as Windows, Linux, MacOS, iOS and Android. Find out more about the latest version of .NET which includes the latest version of C# by visiting dot.net.

Blazor

Blazor allows developers to create beautiful full-stack web applications using the power of .NET and C# without needing to write any JavaScript. Blazor is a modern front-end web framework based on HTML, CSS, and C# to help create web applications quickly using reusable components.

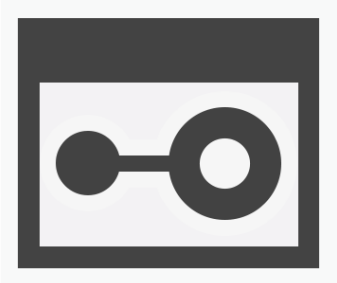
Blazor Components can be run in a browser with WebAssembly or on a server with ASP.NET Core or embedded in native hybrid applications. Install ASP.NET and web development workload for Visual Studio to use Blazor or for ASP.NET Core Web Apps and ASP.NET Core Web APIs. Find out more about using Blazor to create beautiful web applications using .NET by visiting blazor.net.

ASP.NET Core Web App

ASP.NET Core is the open-source framework for building web applications that extends the .NET developer platform with tools and libraries for building web applications. ASP.NET Core supports industry standard authentication protocols and performs faster than any popular framework and can be scaled using Docker containers. ASP.NET Core features real-time bi-directional communication including high-frequency messages from server-side code to connected clients with SignalR.

ASP.NET Core applications can be Model View Controller or MVC with Models and Controllers in C# and Views in Razor or combined with Razor Pages. Find out more about ASP.NET Core to create web applications using .NET by visiting asp.net.

ASP.NET Core Web API



ASP.NET Core is also the open-source framework for building REST APIs with .NET using the same platforms and patterns or side-by-side with ASP.NET Core Web Applications. ASP.NET Core has first-class support for HTTPs and built-in support for JSON Web Tokens along with policy-based authorisation to define powerful access control rules.

ASP.NET Core is designed for modern web experiences allowing endpoints to automatically serialise output to correctly formatted JSON without special configuration. ASP.NET Core not only supports Controllers but also Minimal APIs for simplified approach to building fast HTTP APIs with minimal code and configuration. Find out more about ASP.NET Core Web APIs at dotnet.microsoft.com/apps/aspnet/apis.

What is Windows Presentation Foundation?

Overview

Windows Presentation Foundation or WPF allows developers using modern .NET and C# to write applications targeting Windows Desktop from Windows 7 to the latest Windows 11. WPF was first introduced in 2006 as part of the legacy .NET Framework but it is possible to bring those applications up-to-date and take advantage of the latest features of .NET.

Windows Presentation Foundation allows developers to build a WPF Applications natively on Windows with Blazor Hybrid allowing components to be reused across platforms. Install .NET desktop development workload for Visual Studio to create a WPF Application where you can drag-and-drop controls in the design view or compose XAML. Find source code and how to get started with Windows Presentation Foundation or WPF at github.com/dotnet/wpf.

XAML

XAML or eXtensible Application Markup Language is an XML-based declarative markup language that can define user-interface objects along with rotations, animations, or other effects. XAML enables developers to decouple implementation of an application including events, business object and other code in C# from the design and user interface of an application. XAML allows developers to data bind from source values to target properties including converting values to present data to users or allow interaction with data from users.

XAML supports vector-based controls that can be scaled along with media elements such as vector or raster images or lines and shapes with a variety of brushes. XAML supports vector-based controls that can be scaled along with media elements such as vector or raster images or lines and shapes with a variety of brushes. XAML makes it possible to completely customise the look-and-feel of an application without impacting the implementation of an application.

Controls

WPF Applications can use a toolbox of standard user interface controls such as Button, CheckBox, ComboBox and TextBox that can be dropped in the Design View or typed in the XAML View that can be styled as needed. WPF Application standard controls and others can be customised completely beyond just simple styling to create a custom appearance and structure of a control or even control the visual behaviour of a control.

WPF Applications can have flexible and responsive layouts such as Grid with fluid resizing of elements or stack elements horizontally or vertically with StackPanel along with many more layout designs. WPF Applications support custom user controls for reusable layout elements and templated controls that support customisation of appearance and behaviour of custom controls. WPF Applications can take advantage of XAML created for legacy WPF for .NET Framework or leverage third-party controls designed for modern WPF with the latest .NET.

Summary

.NET is the open-source cross-platform framework from Microsoft allowing developers to develop applications and services for any device or platform such as Web and Desktop. Windows Presentation Foundation to create WPF Applications with .NET desktop development workload installed targeting Windows desktop from Windows 7 to Windows 11. WPF Applications enable developers to create applications with decoupled user interface from any implementation in code and bring them together with databinding. WPF Applications allows developers to use a variety of standard vector-based controls or create custom user or templated controls for their applications. Developers can leverage functionality from across the .NET ecosystem including Blazor Hybrid to create modern WPF Applications.

About

dot.net
blazor.net
asp.net
dotnet.microsoft.com/apps/aspnet/apis
github.com/dotnet/wpf

tutorialr.com/talks/wpf-with-dot-net
github.com/tutorialr/wpf-with-dot-net-talk