



To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

Windows Presentation Foundation (WPF)

Class Duration

35 hours of live training delivered over 5 days.

Student Prerequisites

- Familiarity with the fundamentals of C# and .NET, at the level of an introductory C#/.NET course
- Developers without this background are welcome in the class, but the course moves quickly and they will likely find the pace quite challenging

Target Audience

This course is designed for developers building Windows desktop applications with WPF and C#. It suits teams starting new WPF projects on modern .NET as well as teams porting existing Windows Forms, MFC, or traditional Win32 applications to WPF.

Description

This course provides hands-on experience with Microsoft's premier UI framework for Windows desktop development. It teaches participants how to best utilize WPF for .NET 10 and .NET Framework 4.8.x, both for new projects and for porting existing Windows Forms, MFC, or traditional Win32 applications to the platform.

WPF remains a fully supported, actively developed part of modern .NET. The course covers the framework from the ground up - XAML, layout, the property and rendering architecture, input and commands, data binding, and the Model-View-ViewModel pattern - before moving into the deep customization features that set WPF apart: styles, triggers, control templates, custom controls, and the theme system, including the Fluent (Windows 11) theming introduced in .NET 9 and expanded in .NET 10.

Learning Outcomes

- Understand the motivation behind WPF, its main features, and how a basic application fits together



To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

- Build user interfaces declaratively with XAML while keeping a clean separation of UI and procedural code
- Choose between the built-in layout panels, create custom panels, and fine-tune layout with margin, padding, alignment, and transforms
- Understand the framework architecture: Windows integration, the property system, and how WPF renders visual elements
- Handle input with the WPF input architecture and consolidate event handlers using commands
- Organize applications with a logical project structure, shared resources, styles, and triggers
- Bind business objects to the UI, make objects binding-aware, and provide type conversions for binding expressions
- Present collections with data templates, sorting, filtering, grouping, and hierarchical (parent/child) binding
- Structure applications with the Model-View-ViewModel (MVVM) pattern for better testability
- Use WPF graphics support: shapes, geometries, fills, and special effects such as opacity and reflections
- Redefine the appearance of built-in controls with control templates, triggers, and the Visual State Manager
- Build reusable custom controls and integrate with the WPF theme system, including Fluent theming on modern .NET

Training Materials

Comprehensive courseware is distributed online at the start of class. All students receive a downloadable MP4 recording of the training.

Software Requirements

- Windows 10 or Windows 11
- Visual Studio 2026 (Community Edition or higher) with the .NET desktop development workload
- .NET 10 SDK
- .NET Framework 4.8.x Developer Pack (for the .NET Framework portions of the course)

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

Training Topics

Introduction

- Motivation behind WPF and its main features
- WPF on .NET 10 and .NET Framework 4.8.x
- Primary tools and classes used to build WPF applications
- How a basic WPF application fits together

Using XAML

- Creating UI independently of procedural code
- Clean separation of code and UI
- XAML syntax and code-behind

Managing Layout

- Choosing between the built-in layout panels
- Creating custom panels
- Fine-tuning layout with margin, padding, and alignment
- Special effects through transforms such as rotation and scaling

Framework Architecture

- How WPF integrates with the Windows operating system
- The dependency property system and property value management
- How WPF renders visual elements to the screen

Input Management

- The WPF input architecture and routed events
- Receiving input from child controls
- Commands and consolidating event handlers across control types

Organizing Your Application

- Developer and designer workflow integration
- Separating designer assets into a logical project structure
- Common property setters with WPF styles
- Managing shared properties in resources
- Visual interactivity with triggers

Data Binding: Basics

- Connecting data to the UI
- Binding controls to each other

To discuss this course and customizations:
Call: +1 434-509-6890 or Email: sales@cloudcontraptions.com

- Making business objects binding-aware
- Type conversions for binding expressions

Data Binding: Collections and Hierarchies

- Visualizing business objects with data templates
- Managing collections of objects in the UI
- Customizing display with sorting, filtering, and grouping
- Binding to hierarchies of collections (parent/child)

Model-View-ViewModel

- Separating UI and visualization from procedural logic
- Structuring applications for better testability
- Responsibilities of the model, view, and viewmodel

Graphics

- Overview of WPF graphics capabilities: shapes and geometries
- Creating custom shapes, fills, and special effects such as opacity and reflections in XAML
- Designing visuals with the Visual Studio XAML designer and XAML Hot Reload

Control Templates: Basics

- Redefining the visual appearance of existing controls
- Retemplating the Button with a new visualization
- Visual behavior with triggers and the Visual State Manager (VSM)

Control Templates: Beyond Buttons

- Templating advanced controls: ListBox, Slider, TreeView, and Tab controls
- Best practices for replacing supported templates
- Providing new visualizations while maintaining procedural behavior

Custom Controls

- Building reusable controls through composition and inheritance
- Building a new control from scratch
- Providing default styles for custom controls
- Integrating with the WPF theme system, including the Fluent (Windows 11) theme and light, dark, and system theme modes on .NET 9 and later