

Contents

- [DevForce Windows Phone App](#)
- [DevForce Windows Store App](#)
- [DevForce Silverlight Application](#)
- [DevForce WinForms Application](#)
- [DevForce n-tier WinForms Application](#)
- [DevForce WPF Application](#)
- [DevForce n-tier WPF Application](#)

DevForce provides a number of project templates to help you get started developing your DevForce application. These templates enhance standard Visual Studio templates to add DevForce-specific assembly references and other files. We'll review the templates available to help you **choose the DevForce project template** that's right for your project.

DevForce tutorial videos and sample application are good places to learn about these templates in detail. A brief description of each template follows.

DevForce Windows Phone App

This template generates a solution with a blank Windows Phone application project, along with a web application project to host the DevForce [EntityServer](#). A DevForce Windows Phone application is always an n-tier application and must issue asynchronous queries to the *EntityServer* to retrieve data. The template primes the solution with the files you need to get started.

DevForce Windows Store App

This template generates a solution with a blank Windows Store application project, along with a web application project to host the DevForce [EntityServer](#). A DevForce Windows Store application is always an n-tier application and must issue asynchronous queries to the *EntityServer* to retrieve data. The template primes the solution with the files you need to get started.

DevForce Silverlight Application

The **Silverlight Application Template** generates a bare bones solution consisting of the minimum two projects: a Silverlight application project and a web application project.

Both projects are primed with references to the DevForce libraries. In the web project, the Global.asax and Web.config are prepared for DevForce and there's a log folder ready to receive the log file, DevForceDebugLog.xml, that captures the application's server-side activity.

Begin by adding an Entity Framework model to the web project. DevForce generates the full .NET entity classes in the web project and automatically adds links to them in the Silverlight application project. Those links ("shortcuts") appear in the "Shared Code" folder.

Continue by building out the Silverlight UI with views and presentation logic.

This template is used in the simplest DevForce tutorials in which you will often see the DevForce entity model added to the web application solution. A [separate model project](#) is generally preferred.

The [Tour of DevForce Silverlight](#) walks you through an example using this template.

DevForce WinForms Application

This template produces a **2-tier solution** with a single project targeting **Windows Forms** technology. The essential elements are in place although all you'll see is an empty *Form1* if you build and run. You could add an Entity Framework model to it, use a synchronous query to get some entities, and throw them onto the form in a DataGridView.

A [separate model project](#) is generally preferred.

DevForce n-tier WinForms Application

This **3-tier** version of the [2-tier WinForms](#) adds a web application project to host the [EntityServer](#). The *Form1* is no more revealing but the basics are in place to deploy a server project separately from the **Windows Forms** client and have the latter pull entities over the wire either synchronously or asynchronously.

DevForce WPF Application

This template produces a **2-tier solution** with a single project targeting **Windows Presentation Foundation (WPF)** technology. The essential elements are in place although all you'll see, if you build and run it, is an empty *MainWindow*. You could add an Entity Framework model to it, use a synchronous query to get some entities, and throw them onto the form in a grid.

A [separate model project](#) is generally preferred.

The [Tour of DevForce WPF](#) walks you through an example using this template.

DevForce n-tier WPF Application

This **3-tier** version of the [2-tier WPF](#) adds a web application project to host the [EntityServer](#). The *MainWindow* is no more revealing but the basics are in place to deploy a server project separately from the **WPF** client and have the latter pull entities over the wire either synchronously or asynchronously.