

## Contents

- [Problem](#)
- [Solution](#)
  - [Troubleshooting](#)
- [Prerequisites](#)

This sample application illustrates using the **ASP.NET authentication profile** feature.

- **Platform:** Silverlight
- **Language:** C#, VB
- **Download:** [ASP.NET profile \(Silverlight\)](#)

## Problem

You'd like a small sample Silverlight application which illustrates use of ASP.NET security and profile features.

## Solution

The sample here is provided if all you need is a simple example to see the basics of how to use the ASP.NET profile feature in your application. The sample is similar to the [ASP.NET membership and roles](#) sample, but also shows how a custom user type can be used which integrates with the ASP.NET Profile.

Notice that the web.config contains an enabled `<profile>` element with several properties defined.

A *CustomUser* class has also been created—it's located in the web project but also linked from the Silverlight project—with properties corresponding to the profile properties. DevForce will automatically populate the custom properties from the profile during login.

The custom type must be defined in both Silverlight and web projects because it will be used on both client and server. You can see this by retrieving the *EntityManager.Principal* after the login completes—it will be a *CustomUser* instance. On the server side, we see this also when fetches or saves are performed. In the *EntityServerQueryInterceptor* and *EntityServerSaveInterceptor* classes in *EntityServerEventsHandler.cs* the custom type is available via the *Principal* property.

## Troubleshooting

- If the membership database is not found or does not contain the user's information, the *IsAuthenticated* value displayed on the page is always *False*. If you perform a Windows login while using Forms authentication this will also occur, since your Windows account is not in the database.
- When trying to login, the **Error using ASP.NET Membership** message will display if SQL Express is not installed or the service has not been started. The remainder of the message will indicate the exact cause.

## Prerequisites

This sample is set up to access a local ASP.NET membership database named aspnetdb on an instance of SQL Express. This database contains membership, role, and profile information for two users: "Manager" and "Employee", and is located in a file, aspnetdb.mdf, stored in the Data folder (adjacent to the CodeCS folder).

If you wish to use SQL Server rather than SQL Express you can do so by attaching the aspnetdb.mdf database to SQL Server and changing the *LocalSqlServer* connection string in the web.config. There are comments in web.config (near the bottom) to show you the necessary change.

The aspnetdb.mdf file used in this sample is copied to the App\_Data folder upon first build (see the pre-build event on the web project). It expects to find the file in the Data folder adjacent to CodeCS. If you move the solution, be sure to move the Data folder as well.

You can use the **ASP.NET Configuration** menu item from the **Project** menu to view and edit membership and role information.