# Message Queue Fundamentals in .NET

**Module 2 Setup Instructions**

In Module 2 we start with an existing website with an unsubscribe function that runs synchronously when the user submits a form. We replace the synchronous processing with a fire-and-forget message pattern using MSMQ, with a console app to process the unsubscribe request.

## Pre-requisites

The demo solution is delivered in Visual Studio 2013.

The solution uses NuGet package restore to load packages during the build.

To enable this in Visual Studio, open *Tools…Library Package Manager…Package Manager Settings* and ensure both options (*Allow NuGet to download missing packages* and *Automatically check for missing packages during build in Visual Studio*) are ticked.

The web project is set up to use IIS (not IIS Express), so you will need IIS installed, or you can change the configuration for the web project.

Messaging is done over MSMQ. You will need to have installed MSMQ (as a Windows component), and then create two queues: **sixeyed.messagequeue.unsubscribe** (non-transactional) and **sixeyed.messagequeue.unsubscribe-tx** (transactional).

The Module 3 Setup Instructions detail how to install and configure MSMQ.

## Running the Solution

Build the solution and navigate to **http://localhost/Sixeyed.MessageQueue.Web/unsubscribe** in a browser window to submit the form and start the unsubscribe workflow.

In the **before** solution, the browser will process the call synchronously and you will see a spinner for 40 seconds.

In the **after** solution the website will post a message to **sixeyed.messagequeue.unsubscribe-tx** and the site returns very quickly. Start the console app (or run the solution in Visual Studio), and the app will receive the message and run the workflow.