This document contains the references and resources to accompany my "Upgrading Your Skillset for SharePoint 2013 App Model Development" session. This is by no means an exhaustive list, but it does contain resources that I found extremely helpful as I sought to come up-to-speed with development and associated concerns pertaining to SharePoint 2013's new App Model.

If you have comments, questions, or other feedback, then I encourage you to visit my blog (http://SharePointInterface.com) or contact me via email (sean@sharepointinterface.com) or Twitter (@spmcdonough). I'd love to hear from you!

Last updated on July 29, 2013

- Sean McDonough

App Model Ramp-Up

- **Apps for SharePoint Overview.** This MSDN page provides an end-to-end walk-through on the new SharePoint 2013 App Model, its characteristics, what it seeks to address, and more. http://tinyurl.com/pfp-appsoverview
- **Microsoft Office Developer Tools for Visual Studio 2012.** If you're going to build Apps for SharePoint 2013, you need the right tools. You can get the tools you need for Visual Studio 2012 here; just use the *Download the tools* link midway down the page. http://tinyurl.com/pfp-vstools
- **Choose the Right API Set in SharePoint 2013.** The new App Model isn't appropriate for every development scenario. The information on this MSDN page will help guide you in selecting the appropriate technology for a particular development challenge. http://tinyurl.com/pfp-appsapi
- **Microsoft SharePoint 2013 App Development.** This book by Scot Hillier and Ted Pattison is a wonderful and concise resource for understanding the App Model development landscape. This book does not teach SharePoint development in general and assumes that readers are already familiar with general (full-trust) SharePoint development and SharePoint in general. http://tinyurl.com/pfp-appdevbook
- **Web Essentials for Visual Studio.** This Visual Studio extension (available from within Visual Studio 2012 itself) provides a number of accelerators and helpers to improve your experience when working with JavaScript, CSS, and other App Model technologies. http://vswebessentials.com/
- **App Approaches to Common SharePoint Customizations.** Microsoft's Richard diZerega put this together to compare how many common full-trust customizations can be achieved (or not) under the new App Model. This is a great article to use as a starting point for researching whether or not something can or should be done with the App Model. http://tinyurl.com/pfp-approaches

HTML and CSS

- **The Definitive Guide to HTML5.** This book by Adam Freeman is an end-to-end reference on HTML5. It is nearly 1000 pages in length and is a good, detailed reference.
  http://tinyurl.com/pfp-html5guide
- **Mozilla Developer Network.** The Mozilla Developer Network (MDN) is an excellent online resource for HTML, CSS, and various other browser-based technology topics. Although w3schools.com hits tend to appear first in search engine results, I've found the content on the MDN to be more accurate and "reputable" overall.
  http://tinyurl.com/pfp-mozilla
- **The Book of CSS3.** This book by Peter Gasston is focused solely on CSS3 and nothing else. If you aren't familiar with CSS, then this isn't the reference to start with. If you are relatively comfortable with CSS and need to understand the specifics of CSS3, though, then I wholeheartedly recommend this resource.
  http://tinyurl.com/pfp-css3book
- **Can I use …** This website is your one-stop shop to determine whether or not a particular technology, CSS capability, HTML implementation, or other feature is available in the browser of your choice. The site is regularly updated and includes a feed that you can follow to track site-related changes.
  http://caniuse.com/
- **CSS3 Cheat Sheet (PDF).** This multi-page PDF provides a fairly solid reference of CSS3. Note, though, that the PDF is dated 2009. Since CSS3 is a spec under revision and something of a moving target, there will be inaccuracies.
  http://tinyurl.com/pfp-css3cheatsheet

Responsive Design

- **Responsive Web Design by Ethan Marcotte.** This is often cited as the article that got the whole "responsive design thing" going. Definitely worth a read to develop some insight into fluid grids, flexible images, and media queries.
  http://tinyurl.com/pfp-responsivedesign
- **CSS Media Queries.** A neat site that helps you see how CSS3 media queries are evaluated in real-time with your browser. Try it from your desktop, your phone, your tablet …
  http://cssmediaqueries.com/
- **Building a Modern Grid System.** Written by Zurb's Jonathan Smiley, I've found this tutorial very useful for basic grid setup in web projects. The final product from the tutorial is lighter weight than a full-blown grid system – just what I've needed in certain circumstances.
  http://tinyurl.com/pfp-buildgrid
- **Responsive Grid System.** One of many grid systems that you can employ in your own projects pretty easily.
  http://responsive.gs/
- **Foundation.** Zurb's Foundation system is a fantastic toolkit for building a responsive website. It's packed with a lot of CSS and JavaScript goodness. Be warned, however, that implementing it as-is in your SharePoint Apps will cause some styling conflicts unless you do some CSS and

JavaScript fix-ups. If you find yourself wondering what to do to clean up page chrome, for example, ping me and I'll share the limited work-arounds I've developed.
http://foundation.zurb.com/

- **Responsive SharePoint.** This CodePlex project by Eric Overfield, D'arce Hess, and Gilda Spencer exists to demonstrate how common responsive frameworks (like Twitter Bootstrap and Zurb Foundation) can be made to work with SharePoint. The project is usable as-is and comes with plenty of documentation and deployment support.
http://responsivesharepoint.codeplex.com/

JavaScript and jQuery

- **jQuery.** When I'm working on a SharePoint-hosted App, I always have a browser window open to the online jQuery API Documentation area. If you use jQuery, are contemplating using jQuery, or simply want to learn more, you'll eventually end up on this site.
http://jquery.com/
- **JavaScript: The Good Parts.** This book by Douglas Crockford was recommended by several friends in the SharePoint community, and it is an excellent way to onboard with JavaScript if you have a solid background in a classical programming language. Although the book is less than 200 pages long, it is *dense*. You'll need to read slowly to incorporate everything being presented.
http://tinyurl.com/pfp-javascript
- **Learning jQuery.** I sat in on a session from Karl Swedberg (one of the authors) several years back, and I liked his manner of presentation tremendously. This book is a great resource and rich in examples of how to incorporate jQuery into your JavaScript programming.
http://tinyurl.com/pfp-jquery
- **TypeScript.** TypeScript (from Microsoft) seeks to provide some familiar classical programming language structure and design-time support on top of JavaScript. If you would like to leverage familiar constructs like classes and interfaces in JavaScript, TypeScript might be up your alley.
http://www.typescriptlang.org/
- **JSLint.** Created by Douglas Crockford (the author of *JavaScript: The Good Parts*), this site is a JavaScript code analysis tool. You supply code that you've written, indicate which rules you want to run, and the site evaluates your code for compliance/fitness.
http://www.jslint.com/

JavaScript Libraries

- **Modernizr.** This library can be used to detect the HTML5 and CSS3 capabilities available in a user's browser. The analysis results can then be used to enable certain features (in your code) and enhance the user experience.
http://modernizr.com/
- **jQuery UI.** This library contains a set of interactions, widgets, effects, and utilities to improve the UI/UX of web applications. It provides tools like date pickers, progress bars, autocomplete textboxes, and more. It also allows you to support capabilities like drag-and-drop and resizing in

your web interfaces.
http://jqueryui.com/

- **Knockout.** This library is invaluable for building web applications that leverage the MVVM (Model-View-View Model) pattern. The Knockout site includes some excellent tutorials, and using this library helps avoid a lot of spaghetti code that would naturally arise in normal JavaScript development.
  http://knockoutjs.com/
- **JsRender.** Building output strings and HTML leads to a lot of tight-coupling and spaghetti code. JsRender provides a template-driven approach to avoid this scenario and keep your code easy-to-read.
  http://tinyurl.com/pfp-jsrender
- **Google's Cookie Library.** Cookies are important, especially in SharePoint-hosted Apps where you need to track SharePoint values/URLs across postbacks. Google supplies a very simple library that makes it easy to read and write cookies with minimal hassle.
  http://tinyurl.com/pfp-cookies
- **HTML5 Cross Browser Polyfills.** If you're leveraging newer HTML5 and CSS3 capabilities in your applications and need to support older browsers, you'll likely need the support of polyfills. This list is a great place to find the support you need.
  http://tinyurl.com/pfp-polyfills

CSOM and REST

- **Data access options for apps in SharePoint 2013.** This MSDN page is the launch point for a number of other pages that allow you to determine which libraries, techniques, and calling conventions you can (or should) be using in your Apps.
  http://tinyurl.com/pfp-webcalls
- **How to: Complete basic operations using SharePoint 2013 REST endpoints.** REST is commonly used in SharePoint-hosted Apps and in other JavaScript-based access scenarios. This MSDN page covers many common list and library data interaction scenarios from a REST perspective.
  http://tinyurl.com/pfp-rest
- **How to: Complete basic operations using JavaScript library code in SharePoint 2013.** If you prefer to leverage something closer to the client-side object model in your JavaScript code, then this page covers common list and library data interactions from that perspective.
  http://tinyurl.com/pfp-jsom
- **How to: Complete basic operations using SharePoint 2013 client library code.** If you're building autohosted or provider-hosted Apps, you'll likely be able to leverage the CSOM managed code assemblies to access and work with data stored in SharePoint. This MSDN page covers the basics of how to use the .NET client object model in those assemblies.
  http://tinyurl.com/pfp-csom
- **App Designs in SharePoint 2013.** Scott Hillier put together a great blog post that helps those who are new to Apps to differentiate between App types, data access options that are available, and how they can be combined.
  http://tinyurl.com/pfp-appdesign

- **Open Data Protocol (OData).** The SharePoint REST interface is (mostly) OData compliant. Familiarizing yourself with the OData protocol and its conventions will help you as you learn to create REST queries for SharePoint.
  http://www.odata.org/