You’ve heard of SPA – “single page application” – and wondered what it is. While you could read about it, you’d rather experience it for yourself. But who has time to download a sample? Well if you’ve got Visual Studio, you’ll have a SPA up and running in less than 60 seconds with the Breeze MVC SPA Template.

[Relationship to the ASP.NET MVC SPA Template]

# File | New

[Generate “Zephyr” in the shortest possible steps to Ctrl-F5]

# Run it

[short tour of Zephyr running .. .maybe a video of it running]

# What’s the point?

Zephyr introduces you to many techniques of end-to-end, Breeze-backed SPA design under the umbrella of ASP.NET MVC

* List them

The glaring omission is screen navigation. The “SP” in SPA refers to the single web page that hosts the client application. The client app itself features fast navigation among multiple screens, governed by JavaScript and HTML in the browser. Such an app asks the server for fresh data, not HTML or workflow.

You’ll want to look elsewhere [#link to notes with links to resources] for examples and guidance on this essential characteristic of JavaScript applications.

But there’s plenty to learn from this simple example. Let’s step inside.

# Solution overview

The Breeze SPA template generates a single-project, web application solution with structure and contents that hew closely to the ASP SPA template.

That is entirely intentional. A primary goal of the Breeze template is to demonstrate ***how little you must change*** to enter the more capable world of Breeze development. For the most part we’ll concentrate our attention on those changes. But we can’t do that in a vacuum. So we’ll cover the commonalities as we go, often in cursory fashion leaving the details to others [#link to notes with links to resources].

|  |  |
| --- | --- |
|  | **App\_Data** – The *Todo* LocalDb database files; “Show all files” to see them.  **App\_Start** – Web application configuration classes  **Areas** – MVC location for Web API help  **Content** – CSS and theme images  **Controllers** – both MVC page and Web API data controllers  **Filters** – A Web API attribute applied to the AccountController that can create and initialize the database if not found. |