

DDD with ASP.NET Core – Labs

Steve Smith (@ardalis)

May 2017

Lab 0: Set up

Goal

Have a working development environment and starter application.

Requirements

Visual Studio 2017 with .NET Core Workload or **Visual Studio Code plus CLI tools**. A git client. **Note: Not VS2015.**

It will also be helpful to install:

- **Postman** or **Fiddler** or another tool that allows you to GET/POST/PUT/DELETE to HTTP endpoints
- **Smtplib4Dev** or **Papercut** or another local email testing tool

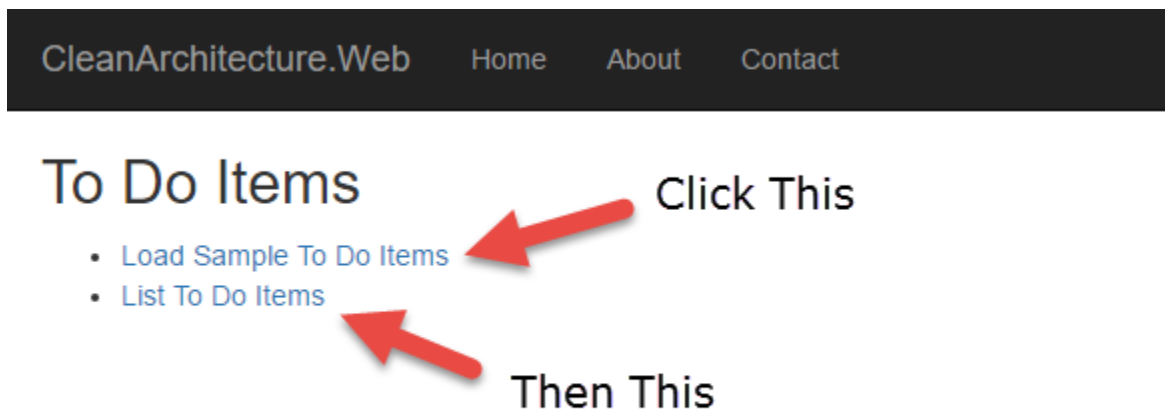
Instructions

We're going to be building a Guestbook application today using ASP.NET Core, Domain-Driven Design principles and patterns, and a domain-centric architecture. It's important that the solution be structured properly to support this approach. We will be starting from a Clean Architecture template which you'll find here:

<https://github.com/ardalis/CleanArchitecture>

Clone this sample to a working folder on your machine. If you download a ZIP of the repository, be sure to **Unblock** the zip file before you extract its contents (right click, Properties, Unblock).

You should be able to run the application and view the web app in your browser (F5 in Visual Studio; **dotnet restore / dotnet run** from terminal). You should be able to populate and view several ToDo items (see image below):



You should be able to run the tests in the tests folder, either from the command line or a test runner within your IDE. From command line, go to test project folder, **dotnet restore / dotnet test**. They should all pass.