The .NET Core MVC & .NET Core Razor Pages Story – aka “The COMPLETE Story” Workshop

Agenda

Chapter 1

For this to be the “complete” story as the title of this workshop eludes to, first, I will pay homage by starting with the history of ASP.NET and Microsoft’s support of MVC.

Chapter 2

Fundamentals of MVC: Model, View, Controller, View Model and many more.

History of Microsoft support of the various tooling, options of MVC

Chapter 3

Now for something FRESH… We will dive into an in-depth comparison of .NET Core MVC & .NET Core Razor Pages.

Chapter 4

Razor Pages

Chapter 5

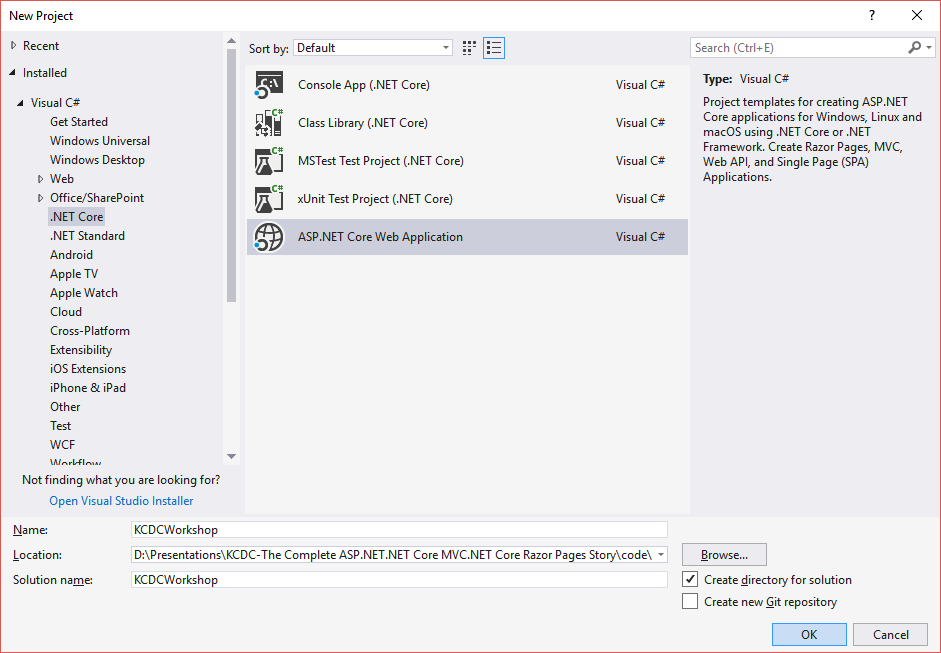
Available Resources

Chapter 6

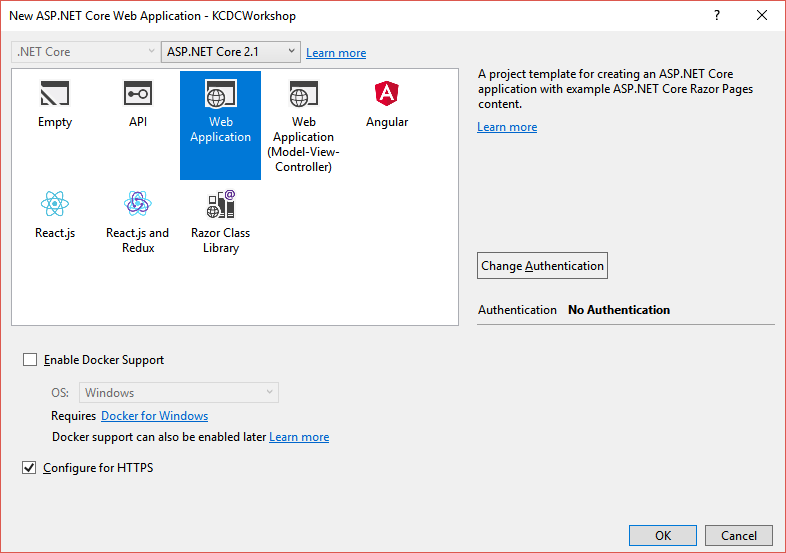
Hands- On

The .NET Core MVC & .NET Core Razor Pages Story Workshop – Step by Step - Hands-On

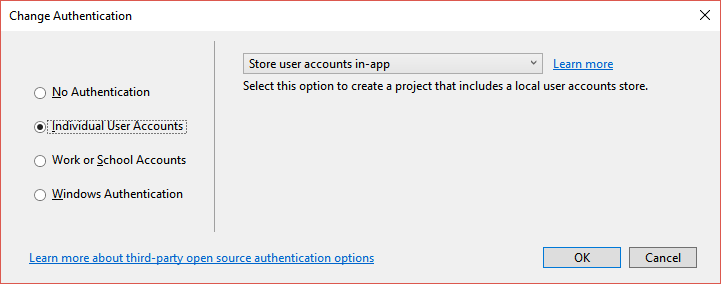
Step 1. Open Visual Studio and create new Project - ASP.NET Core Web Application.



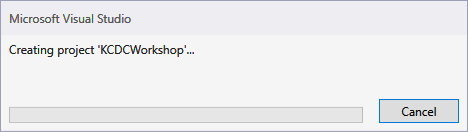
2. Explore all the options, but for this exercise select “Web Application” (with Razor).



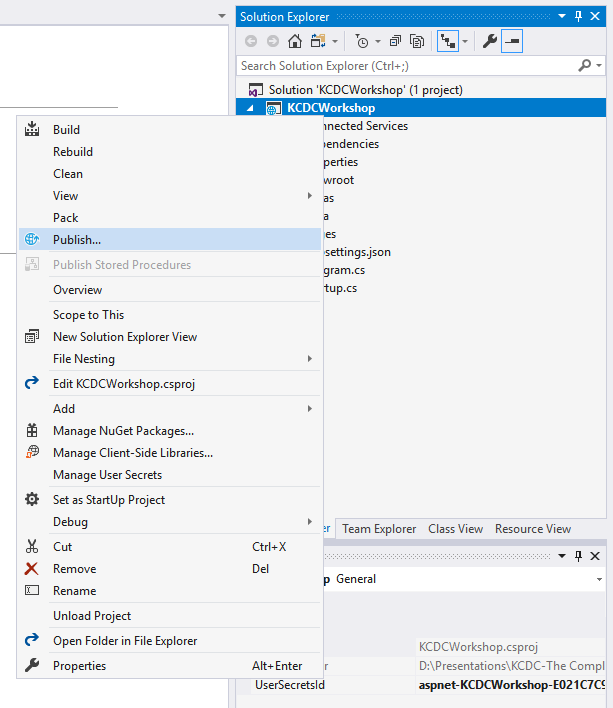
3. Click “Change Authentication” and select “Individual User Accounts”. Click Ok.



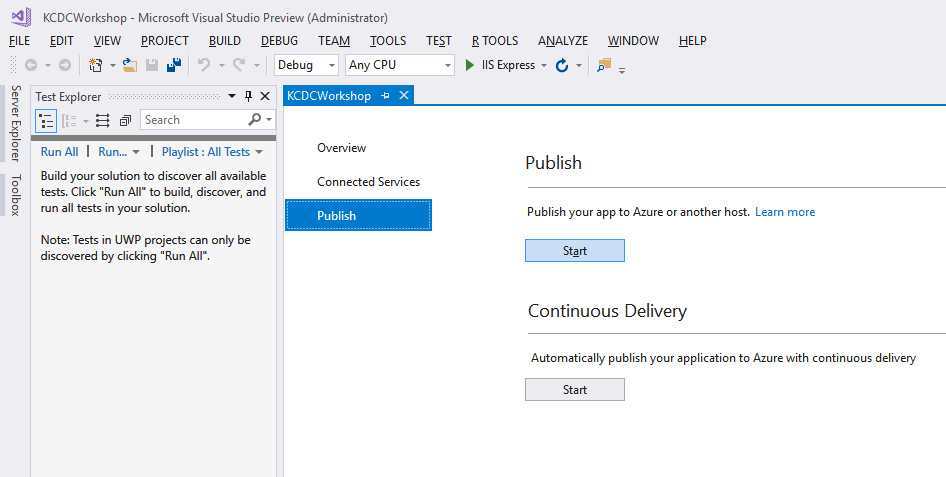
4. Click Ok again on “New Project” window and we are off to the races.



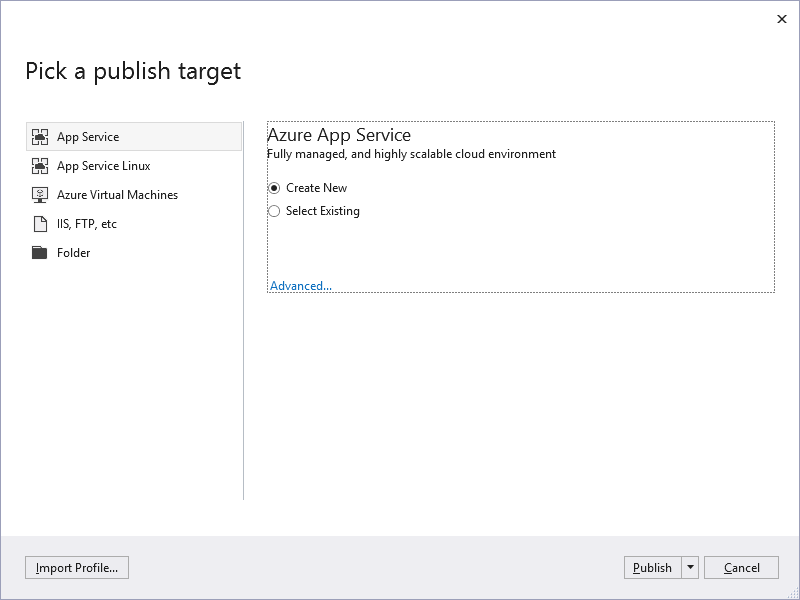
5. Co-workers don’t let co-workers right click Publish. (Err., but wait, now you can!)



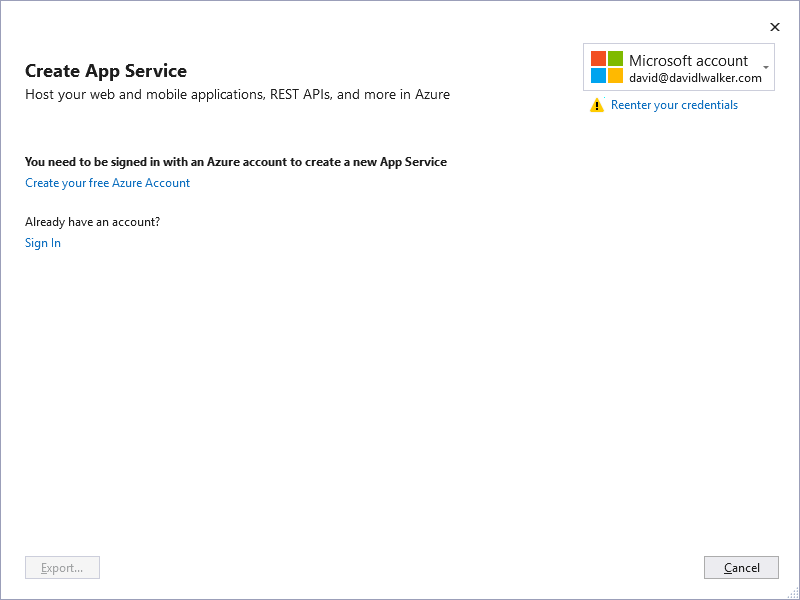
6. Click Start button under Publish.



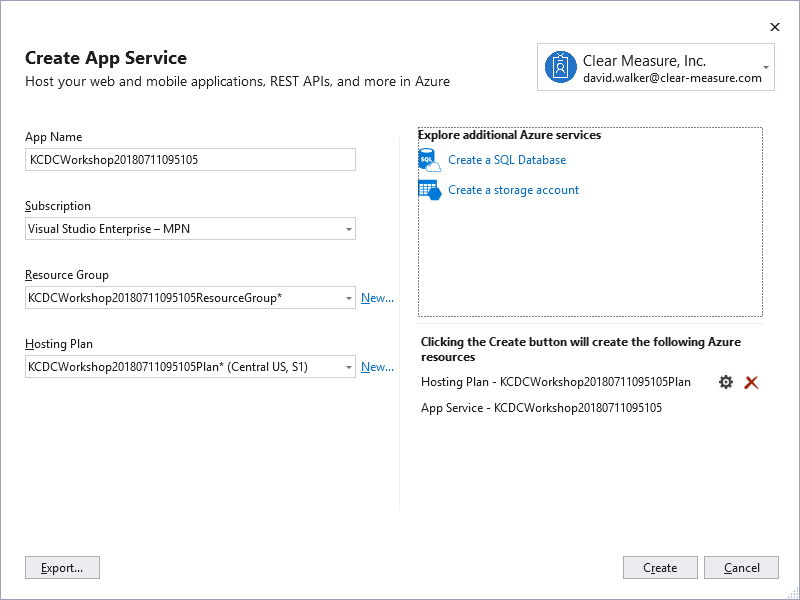
7. Pick a Publishing Target. Leave the default “Create New” and click Publish.



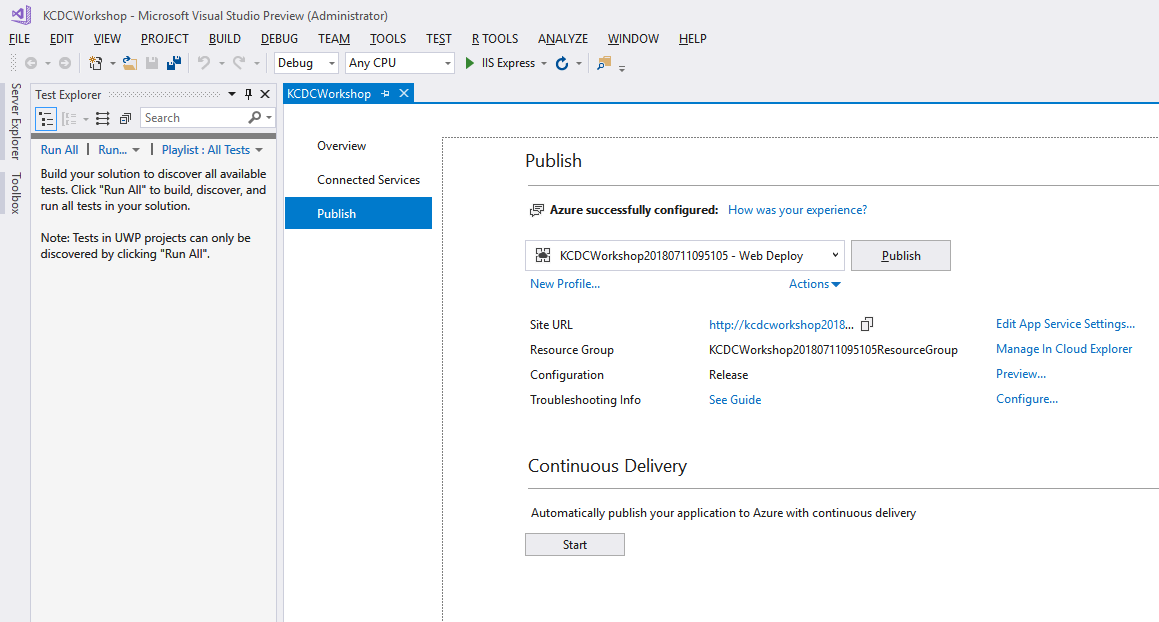
8. Enter Azure Credentials.



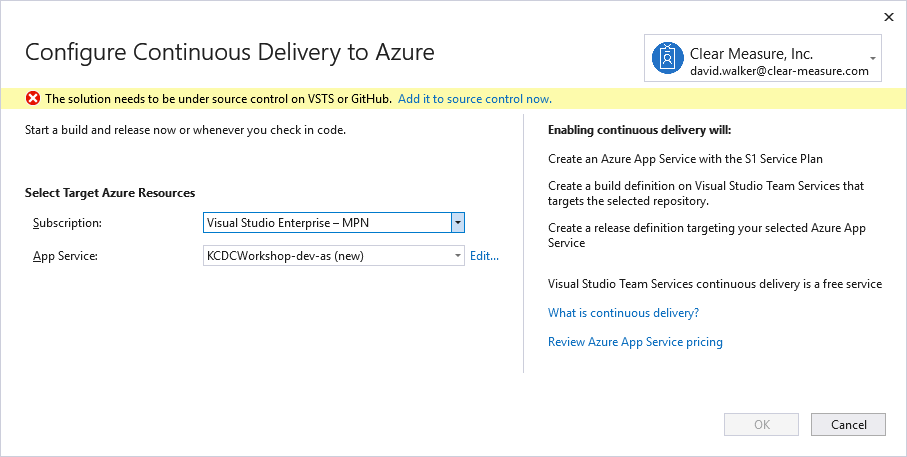
9. Select a Subscription. Leave the rest default. Click Create button.



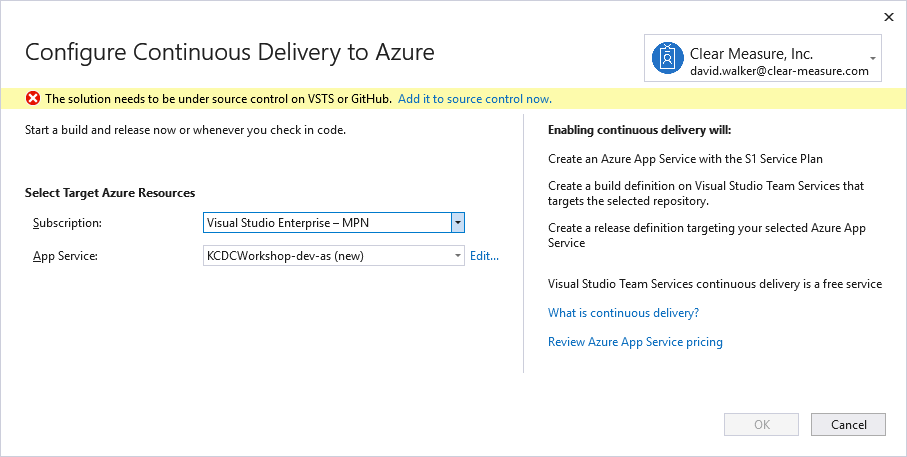
10. Azure successfully configured. – Next click Start under Continuous Delivery.



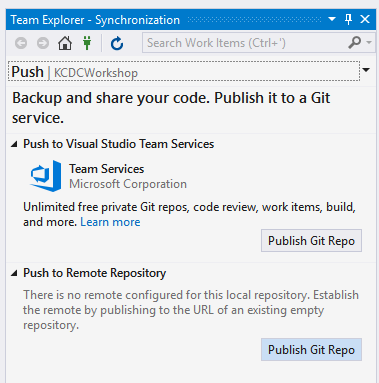
11. Continuous Delivery. – You’ll want to



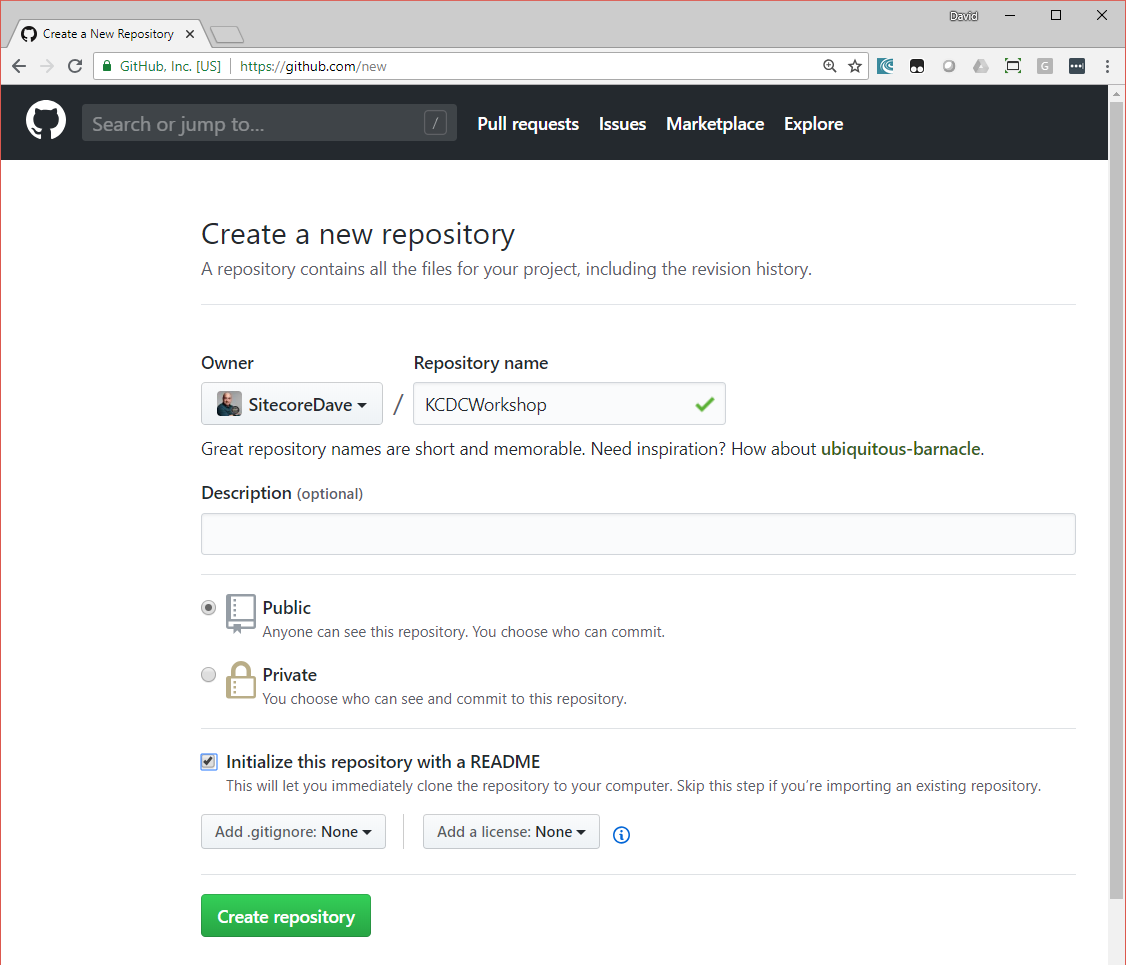
12. Click “Add it to source control now”



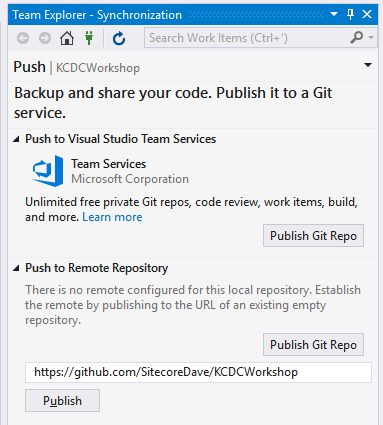
13. Choose your repo.



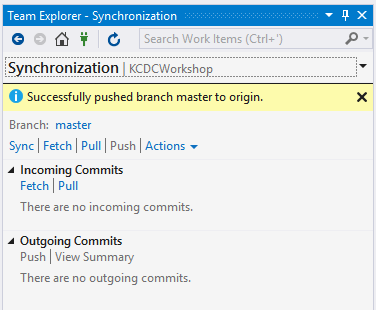
14. Github.com



15. Enter repo location in Visual Studio. Click Publish.



16. Successfully pushed branch master to origin.



17. Configure Continuous Deployment.

