# About this book

This book has been a culmination of long-time vision, some key leadership, and a confluence of industry events. Almost fifteen years ago, the author gained a passion for helping developers succeed, for making the complex simple, and for finding rules of thumb that would work for 80% of situations. With too many options in the software world and too many answers of “it depends”, the industry has been starved for the ability to do something “by the book.” This book seeks to provide that text where a .Net developer can say “I’m doing DevOps with .Net and Azure by the book.” In this manner, one would know what models and patterns were in play and what to expect from said environment. This book is being released while .Net Core 3 is in preview status; therefore, the version of the book should be considered preview as well. The examples largely use Visual Studio 2019 preview edition. The code itself and the Azure DevOps Services pipeline functions perfectly well with .Net Core 2.2, however, and can be used to implement applications immediately. It is the intent of the author to release a .Net Core 3 edition aligning with Microsoft’s release schedule. The example configuration used throughout this book can be leveraged through a public project and source code repository online at https://dev.azure.com/clearmeasurelabs/Onion-DevOps-Architecture.

# Acknowledgements

First, I must thank God and his son, Jesus, for giving me the ability to think and write. Next, I’d like to thank my beautiful wife, Liana, for being awesome at her job and affording me the flexibility to go away and concentrate for long periods of time while writing this text. With three kids in the household, she is an expert mother and homemaker, and this book would not exist without her expert work. Thanks also to my parents, Peter and Rosemary Palermo, for instilling in me a love of books and learning from an early age. I also need to recognize my college professor at Texas A&M, Mike Hnatt, who, through his programming courses, business coaching, and ongoing friendship, has continued to mentor me. Additionally, I’d like to acknowledge Jack Welch, of whose MBA program I am a graduate. From him, his books, and his curriculum, I learned to use fewer words when presenting ideas and information.

To Steve Hickman, thank you for being my first and longest-tenured software mentor. Steve was my first boss. He hired me for my first programming position twenty-two years ago. He now mentors the software engineers at architects at Clear Measure, Inc. He has been instrumental in forming the vision for how to simplify software engineering on the .Net platform.

To Megan Beutler, thank you for your ongoing encouragement. Megan has been a part of Clear Measure for over five years and has been a constant source of encouragement and positivity. She brightens the day of anyone in her path.

To the engineers and architects at Clear Measure, Inc., who are blazing the trail of better .Net software methods and the DevOps approach illustrated in this book. Without the learning from the many client projects, the guidance of this book would not exist. Thank you to: Megan Beutler, Rayne Fulton, Tony Fauss, Nick Becker, Scott Wilson, Troy Vinson, Danny Vandergriff, Vlad Serafimov, Eric Fleming, Colin Pear, Corey Keller, Kyle Nunery, Steve Hickman, Eric Williams, Mike Alpert, Mike Sigsworth, Zeeshan Ansari, Joe Lockbaum, Valerie Gurka, Haley Akchurin, Trish Polay, Eric Farr, Richard Hartness, Monica Pritchard, Jim Wallace, Justin Basinger, and Chris Thomas.

To Scott Guthrie, whose leadership at Microsoft not only with .Net but also with the Azure platform has made it the leading computing platform on the planet. I’ve considered Scott a friend since 2006. In 2017, I met with Scott at his office for almost an hour talking about the problem that is addressed in this book. That most .Net developers have too many options, too many disconnected tools, and that the DevOps environment for .Net and Azure isn’t defined in a cohesive, simple way. Scott encouraged me to build the model that would show developers how to bring everything together.

To Scott Hunter, who heads up the .Net platform at Microsoft. Scott and I discussed this book in mid-2018 over lunch, mulling over the complexity of choices developers need to make when choosing how to bring together the different elements of the Microsoft platform in order to create an environment that causes the “pit of success” – where things fall into place because everything is integrated properly. Thank you, Scott, for being the sponsor of this book.

Finally, I’d like to mention some influential people who have taught me, knowingly or not, some key skills and habits along my career. Each of these has had either a direct or indirect impact on the synthesis of ideas and patterns in this book. First, Robbie McDonald, who took a chance on a cold resume for an internship at Dell. Through the chance he took on me, I was exposed very early in my career to very complex, sophisticated, and high scale computer systems. Next, Eric Brand was the first architect I worked under. At that time, I didn’t understand the difference between a software developer and a software architect. Eric was patient, encouraging, and always had time for questions. Under his projects, I became deeply adept with the .Net runtime and SQL Server, much more so that I had been in the past. When the Iraq war started in 2003, my Army Reserve unit was called up for back-to-back tours in Iraq for 15 months. I was paired with Brett Rogers as co-truck-commanders in our HET tank transport company. Brett was also a .Net developer. Thanks, Brett, for the welcome .Net pair programming out in the Iraq desert. Next, I’d like to acknowledge Steve Donie. Steve taught me continuous integration and the mechanisms of build scripts. The build script in this book is an adaptation of build scripts he authored many years ago in both Ant and NAnt. Additionally, Jeremy Miller taught me dependency management and test-driven development. As the author of StructureMap, Jeremy afforded me an opportunity to contribute to that open-source project as well as the deep learning applying it on some complex software projects. Paul Leury also played a key role in some of the patterns in this book. Paul hired me into my first management position. The team I managed under Paul built a significant native Windows desktop application. During that project, my thoughts on hub & spoke architecture and application buses solidified. The learning on that project made its way into the application bus pattern in the MvcContrib open source library as well as some projects that contributed to Jimmy Bogard’s Mediatr library. Code in this example application is adapted from both of those open source projects. I next must thank Eric Hexter and Stephen Balkum for hiring me as an independent consultant advising one of their team architects, Blake Caraway, who has remained a good friend. Through that work, the guidance on team workflow and process emerged and. Additionally, Eric has been a thought leader on deployment automation and production operations. I learned a great deal in those areas from him. He was a terrific asset as a co-founder of the MvcContrib OSS project as well as a co-author with me on previous books in the “ASP.NET MVC in Action” series. Some of his work is cited in this book. Additionally, I thank my previous co-authors of various books. In working with them I became better at authoring larger bodies of work. Thanks to Ben Scheirman, Jimmy Bogard, Jeremy Skinner, Matt Hinze, and Eric Hexter. Finally, thanks to all the great people in the Microsoft MVP program. I’ve learned something from each of you since my time in the program from 2006 until now. Specifically, thanks to Carl Franklin, for helping me start the Azure DevOps Podcast and for your friendship.

My final acknowledgement goes to Kevin Hurwitz. We made the transition from developer to architect at a similar time. Kevin was instrumental as a sounding board, challenger, and co-professional with me as we developed some of the ideas in this book while working on mutual projects. Notably was when Kevin introduced me to the SQL schema migration pattern that has now become industry standard and implemented by multiple libraries and products. The pattern he taught me is now recognized across the DevOps community and is in this book. Kevin has also contributed to many of the other ideas that have made their way into this book. Thank you, Kevin, for your ideas, innovation, and your friendship over the years.