1. Record Nr. UNINA9910616383903321

Autore	Cashmore Stacy
Titolo	Beginning Azure static web apps : building and deploying dynamic Web applications with Blazor / / Stacy Cashmore
Pubbl/distr/stampa	New York, New York : , : Apress, , [2022] ©2022
ISBN	1-4842-8146-2
Edizione	[First edition.]
Descrizione fisica	1 online resource (296 pages)
Disciplina	004.6782
Soggetti	Web applications Application software - Development Cloud computing
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Intro -- Table of Contents -- About the Author -- About the Technical Reviewers -- Acknowledgments -- Part I: Getting Started -- Chapter 1: Introduction -- JamStack -- BlamStack -- Azure Static Web Apps -- Book Focus -- Let's Get Started! -- Chapter 2: Setting Up the Environment -- Microsoft Account and Azure Subscription -- GitHub Account -- GitHub Repo -- Naming the Repository -- Add a README File -- Add a .gitignore -- Choose a License -- Default Branch -- Git -- Clone the Repository to Our Machine -- Visual Studio 2022 -- Other Languages -- Conclusion -- Chapter 3: Creating the Application -- Technical Requirements -- A Brief Introduction to Git -- Clone -- Pull -- Commit -- Push -- Creating the Client Application -- Exploring the Scaffolded Blazor App -- Application Root Folder: Client -- Program.cs -- _Imports.razor -- App.razor -- wwwroot Folder -- CSS Folder -- index.html -- Shared Folder -- MainLayout.razor -- NavMenu.razor -- Pages Folder -- Index.razor -- Counter.razor -- FetchData.razor -- Running the Application -- Adding the API Application -- Creating the Azure Function Project -- Consuming the Function in the Client App -- Create a Function to Deliver the Weather Forecast -- Call the Function from the Client Application -- Pushing to GitHub -- Conclusion -- Chapter 4: Creating the Static Web App -- Technical Requirements -- Create Resource

Group -- Create the Static Web App -- Base Information -- Logging In to GitHub -- Setup for the Blazor Build -- Opening the New Resource -- Overview of the GitHub Workflow -- Viewing the Deployed Application -- Conclusion -- Chapter 5: Simple Debugging -- Technical Requirements -- Local vs. Azure Environment -- Visual Studio Debugging -- Setting the Port Numbers -- Changes to the Client Project -- Changes to the Api Project -- Running the Application -- Debugging the Application.
Static Web App CLI Debugging -- Resetting the Code -- The Azure Static Web App CLI -- Install Azure Functions Core Tools -- Install NPM -- Installing the Static Web App CLI -- Running the Static Web App CLI -- Debugging the Client Using the CLI -- Cleaning Up the Breakpoints -- Save the Changes for the Port Numbers -- The CLI and Real Azure Resource -- Conclusion -- Part II: Creating Our App -- Chapter 6: Retrieving Blog Data -- Technical Requirements -- Clean Up -- Api -- Client -- Creating the Data Source -- Azure CosmosDB Free Tier -- Create the Azure Resource -- Seed the Table -- Blog Post Retrieval -- Data Model Classes -- Retrieving Blog Post Summaries -- Retrieving a Single, Full Blog Post -- Deploying the Functions -- Conclusion -- Chapter 7: Displaying Data -- Technical Requirements -- Reading the Summaries -- Referencing the Models Project -- Create the Service -- Creating the Blog Post Summary Page -- Display the Whole Summary -- Displaying the Whole Blog Post -- Reading the Blog Post -- Display Blog Post -- Add the First Blog Post to the Index Page -- Prepare the Index.razor File -- Create the BlogPostSummary Component -- Change the Blog Post Summary Page -- Checking Our Results -- Running Locally -- Deploy Blog Post -- Conclusion -- Chapter 8: Static Web App Configuration -- Technical Requirements -- Creating the staticwebapp.config.json File -- Navigation Fallback -- Routes -- Restricting Access to Resources -- Protecting Resources -- Moving Resources - Temporarily -- Moving Resources - Permanently -- Response Overrides -- Example Configuration -- Conclusion -- Part III: Authentication -- Chapter 9: Authentication -- Technical Requirements -- Authentication Basics -- Standard Authentication -- ClientPrincipal -- identityProvider -- userId -- userDetails -- userRoles -- Accessing Authentication Data in Our Application.
Adding API Authentication -- Create the Authentication Library -- Add Authentication Library to the Api Project -- Client-Side Authentication -- Add Authentication Library to Client Project -- Login Screen -- Role-Based Access -- Conclusion -- Chapter 10: Creating Blog Posts -- Technical Requirements -- Adding the Azure Functions -- Creating Blog Posts with the POST Method -- Updating Blog Posts with the PUT HTTP Method -- Deleting Blog Posts with the DELETE HTTP Method -- Securing the Functions -- Updating the Blog Post Services -- Creating Blog Posts -- BlogPostService Changes -- BlogPostSummaryService Changes -- Updating Blog Posts -- BlogPostService Changes -- BlogPostSummaryService Changes -- Deleting Blog Posts -- BlogPostService Changes -- BlogPostSummaryService Changes -- Adding the Blog Post Edit Page -- Adding the Create Link -- Adding the Edit and Delete Links -- Deploy and Test -- Conclusion -- Part IV: SWA Functionality -- Chapter 11: Static Web App CLI -- Technical Requirements -- Running the Application -- Changing Port Numbers -- Azure Function Port Numbers -- Changing the Static Web App CLI Default Port -- Changing the Client Port -- Running Static Content -- Publish the Client Project -- Run the Static Web App CLI with Static Files -- Extending the .gitignore File -- Static Web App CLI Configuration -- Commit Our Changes -- Conclusion -- Chapter 12: Testing in Azure -- Technical Requirements -- Azure Static Web App Staging Slots --

Making a Branch -- Creating the Staging Slot -- Testing in the Cloud -- Making a Second Change -- Revisiting the Stage Environment -- Staging Environment Settings -- Staging Environment Limitations -- Free Tier -- Number of Staging Slots -- Public Access to Staging Slot -- Removing Staging Environments -- Automatically Cleaning Staging Environments -- Manually Cleaning Up Environments -- Conclusion. Chapter 13: Custom Domains -- Technical Requirements -- Adding a Custom Domain -- What Domains to Add -- Azure Portal Custom Domain Pane -- Self-Managed Domain -- Domain on Azure DNS -- Azure-Generated URL -- Conclusion -- Appendix A: Next Steps -- Add User Details -- Blog Post Summary Refresh -- Search -- Custom Summary Length -- Preview Complete Flag -- Tag Lookup -- Blog Post Preview -- Make the Application Multiauthor -- Fully Implement Post Status -- Styling -- Appendix B: Changing the GitHub Workflow -- Opening the Workflow File -- Index.

Sommario/riassunto

Create rich and dynamic web applications on the Azure cloud platform using static web development techniques built around Blazor WebAssembly, APIs, and Markup, while leveraging the paradigm commonly known as JAMstack. This book starts off showing you how to create an environment for deploying your first application. You will create an Azure Static Web App using a Blazor WebAssembly application and adding dynamic content using an Azure function before deploying from GitHub. You will learn to debug your Static Web App locally, both inside of Visual Studio and from the command line using a simple Static Web App CLI command. The book takes a deep dive into the CLI to allow you to emulate all of the features available in the Azure environment. You will learn the authentication and authorizing options with your app and create new blog posts with the post creation function. Included in the book is setting a custom domain and discussion of the options. The book also explores the differences between the free and standard hosting tiers for Static Web Apps. After reading this book, you will be able to create Azure Static Web Apps using Blazor technology.