XAMARIN.FORMS

SUCCINCTLY

BY **ALESSANDRO DEL SOLE**

SUCCINCTLY EBOOK SERIES

Syncfusion®

Xamarin.Forms Succinctly

By

Alessandro Del Sole

Foreword by Daniel Jebaraj

Syncfusion®

Deliver innovation with ease®

2501 Aerial Center Parkway
Suite 200
Morrisville, NC 27560
USA

ISBN: 978-1-64200-212-6

Technical Reviewer: James McCaffrey
Copy Editor: Courtney Wright
Acquisitions Coordinator: Tres Watkins, VP of content, Syncfusion, Inc.
Proofreader: Jacqueline Bieringer, content producer, Syncfusion, Inc.

Table of Contents

The Story Behind the *Succinctly* Series of Books

Daniel Jebaraj, Vice President
Syncfusion, Inc.

Staying on the cutting edge

As many of you may know, Syncfusion is a provider of software components for the Microsoft platform. This puts us in the exciting but challenging position of always being on the cutting edge.

Whenever platforms or tools are shipping out of Microsoft, which seems to be about every other week these days, we have to educate ourselves, quickly.

Information is plentiful but harder to digest

In reality, this translates into a lot of book orders, blog searches, and Twitter scans.

While more information is becoming available on the internet and more and more books are being published, even on topics that are relatively new, one aspect that continues to inhibit us is the inability to find concise technology overview books.

We are usually faced with two options: read several 500+ page books or scour the web for relevant blog posts and other articles. Just as everyone else who has a job to do and customers to serve, we find this quite frustrating.

The *Succinctly* series

This frustration translated into a deep desire to produce a series of concise technical books that would be targeted at developers working on the Microsoft platform.

We firmly believe, given the background knowledge such developers have, that most topics can be translated into books that are between 50 and 100 pages.

This is exactly what we resolved to accomplish with the *Succinctly* series. Isn't everything wonderful born out of a deep desire to change things for the better?

The best authors, the best content

Each author was carefully chosen from a pool of talented experts who shared our vision. The book you now hold in your hands, and the others available in this series, are a result of the authors' tireless work. You will find original content that is guaranteed to get you up and running in about the time it takes to drink a few cups of coffee.

Free forever

Syncfusion will be working to produce books on several topics. The books will always be free. Any updates we publish will also be free.

Free? What is the catch?

There is no catch here. Syncfusion has a vested interest in this effort.

As a component vendor, our unique claim has always been that we offer deeper and broader frameworks than anyone else on the market. Developer education greatly helps us market and sell against competing vendors who promise to “enable AJAX support with one click,” or “turn the moon to cheese!”

Let us know what you think

If you have any topics of interest, thoughts, or feedback, please feel free to send them to us at succinctly-series@syncfusion.com.

We sincerely hope you enjoy reading this book and that it helps you better understand the topic of study. Thank you for reading.

Please follow us on Twitter and “Like” us on Facebook to help us spread the word about the *Succinctly* series!

Note on the Fourth Edition

This is the fourth edition of *Xamarin.Forms Succinctly* by Alessandro Del Sole. This update covers features added from Xamarin.Forms version 3.5 to version 5.0, the XAML code editor, tools for developing iOS applications, and using native Android and iOS controls without writing custom APIs.

About the Author

Alessandro Del Sole is a Xamarin Certified Mobile Developer and has been a Microsoft MVP since 2008. Awarded MVP of the Year in 2009, 2010, 2011, 2012, and 2014, he is internationally considered a Visual Studio expert and a .NET authority. Alessandro has authored many printed books and ebooks on programming with Visual Studio, including *Visual Studio 2017 Succinctly*, *Visual Basic 2015 Unleashed*, and *Visual Studio Code Succinctly*. He has written tons of technical articles about .NET, Visual Studio, and other Microsoft technologies in Italian and English for many developer portals, including MSDN Magazine and the Visual Basic Developer Center from Microsoft. He is a frequent speaker at Italian conferences, and he has released a number of Windows Store apps. He has also produced a number of instructional videos in both English and Italian. Alessandro works as a senior software engineer for Fresenius Medical Care, focusing on building mobile apps with Xamarin in the healthcare market. You can follow him on Twitter at @progalex.

Introduction

For mobile app developers and companies that want to be represented on the market by mobile applications, the need to publish Android, iOS, and Windows versions of apps has dramatically increased in the last few years. For companies that have always worked with native platforms and development tools, this might not be a problem. It is a problem, though, for companies that have always built software with .NET, C#, and, more generally, with the Microsoft stack. A company might therefore hire specialized developers or wait for existing developers to attain the necessary skills and knowledge to work with native platforms, but in both cases, there are huge costs and risks with timing. The ideal solution is that developers could reuse their existing .NET and C# skills to build native mobile apps. This is where Xamarin comes in.

In this book, you will learn how Xamarin.Forms allows for cross-platform development, letting you create mobile and desktop apps for Android, iOS, Tizen devices, macOS, and Windows from a single C# codebase, and therefore reuse your existing .NET skills. You will learn how Xamarin.Forms solutions are made, what makes it possible to share code, how to create the user interface, how to organize controls within containers, and how to implement navigation between pages. You will also leverage advanced techniques, such as data binding and accessing native APIs from cross-platform code.

It is worth mentioning that Xamarin.Forms also supports the F# programming language, but C# is clearly the most common choice, and therefore, all the explanations and examples will be provided based on C#. In Visual Studio 2019, Xamarin.Forms only supports UWP for Windows development. For this reason, when I refer to Windows from now on, I mean Windows 10 and the Universal Windows Platform, not the previous versions.

I will assume you have at least a basic knowledge of C# and the Visual Studio IDE. I also suggest you bookmark the official Xamarin documentation portal for quick reference. The source code for this book is available on GitHub. File names are self-explanatory so that it's easier for you to follow the examples, stored under folders whose names match the chapter number. Before you start writing code, you need to set up your development environment. This is the topic of the first chapter.

Chapter 1 Getting Started with Xamarin.Forms

Before you start writing mobile apps with Xamarin.Forms, you first need to understand the state of mobile app development today and how Xamarin fits into it. Also, you need to set up your development environment to be able to build, test, debug, and deploy your apps to Android, iOS, and Windows devices. This chapter introduces Xamarin as a set of tools and services, introduces Xamarin.Forms as the platform you will use, and then presents the tools and hardware you need for real-world development.

Introducing Xamarin and Xamarin.Forms

Xamarin is the name of a company that Microsoft acquired in 2016 and, at the same time, the name of a set of development tools and services that developers can use to build native apps for iOS, Android, and Windows in C#. Xamarin's main goal is to make it easier for .NET developers to build native apps for Android, iOS, and Windows reusing their existing skills. The reason behind this goal is simple: building apps for Android requires you to know Java and Android Studio or Eclipse; building apps for iOS requires you to know Objective-C or Swift and Xcode; and building apps for Windows requires you to know C# and Visual Studio. As an existing .NET developer—whether you are experienced or a beginner—getting to know all the possible platforms, languages, and development environments is extremely difficult, and costs are extremely high.

Xamarin allows you to build native apps with C#, based on a cross-platform, open-source porting of the .NET Framework called Mono. From a development point of view, Xamarin offers a number of flavors: Xamarin.iOS and Xamarin.Mac, libraries that wrap native Apple APIs you can use to build apps for iOS and macOS using C# and Visual Studio; Xamarin.Android, a library that wraps native Java and Google APIs you can use to build apps for Android using C# and Visual Studio; and Xamarin.Forms, an open-source library that allows you to share code across platforms and build apps that run on Android, iOS, and Windows from a single C# codebase.

The biggest benefit of Xamarin.Forms is that you write code once, and it will run on all the supported platforms at no additional cost. As you'll learn throughout this ebook, Xamarin.Forms consists of a layer that wraps objects common to all the supported platforms into C# objects. Accessing native, platform-specific objects and APIs is possible in several ways (all discussed in the next chapters), but it requires some extra work. Additionally, Xamarin integrates with the Visual Studio IDE on Windows and is part of Visual Studio for Mac, so you can not only create cross-platform solutions, but also write code on different systems. If you look at Code Listing 1, you can see an example of XAML code that defines a page, and associated C# code that can define the behavior of the page.

Code Listing 1

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  x:Class="HelloWorld.HomePage">
</ContentPage>

using Xamarin.Forms;
namespace HelloWorld
{
  public partial class HomePage : ContentPage
  {
    public HomePage ()
    {
      InitializeComponent ();
    }
  }
}
```

The companion environment for Xamarin is the Microsoft App Center, a complete cloud solution for the app-management lifecycle that includes everything you need for build automation, internal distribution, analytics with diagnostics, and automated tests on more than 400 device configurations.

This book focuses on Xamarin.Forms and targets Visual Studio 2019 on Windows 10, but all the technical concepts apply to Visual Studio for Mac as well. However, if you prefer working on a Mac, I recommend that you read my book *Xamarin.Forms for macOS Succinctly*. I've also recorded a video series for Syncfusion that provides an overview of what Xamarin offers, and of Xamarin.iOS, Xamarin.Android, and Xamarin.Forms.

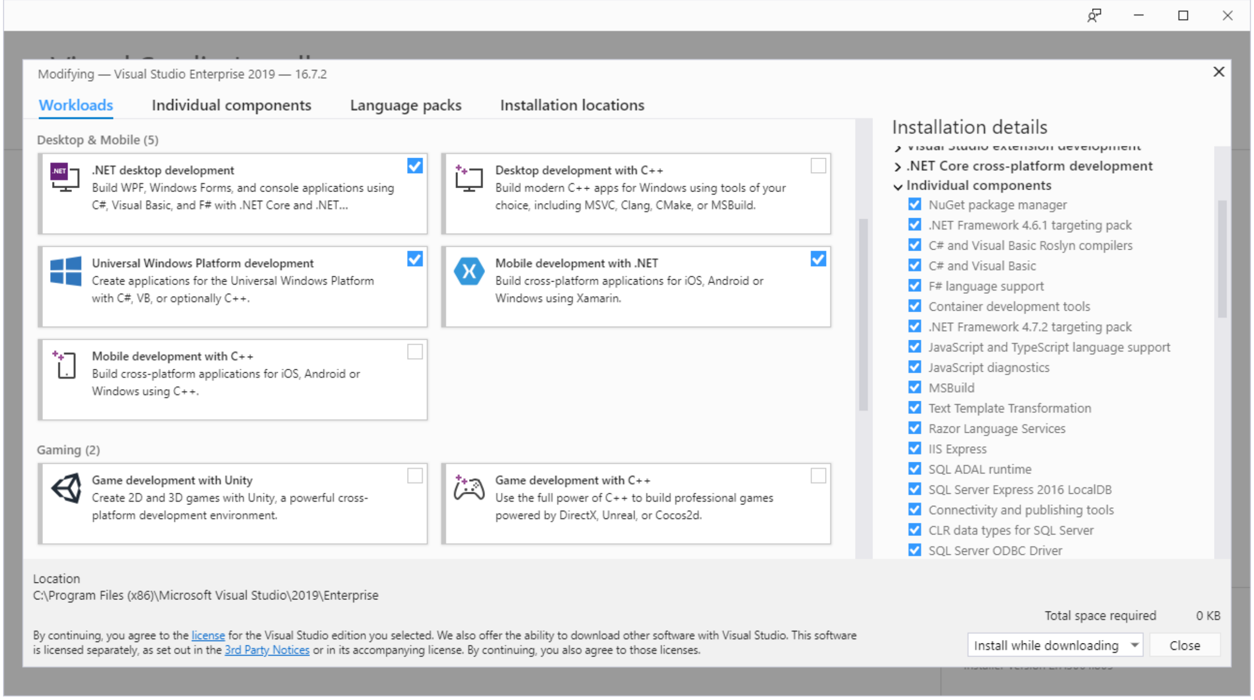
Supported platforms

Out of the box, Xamarin.Forms allows creating apps for Android, iOS, and the Universal Windows Platform from a single C# codebase. Recently, the range of supported platforms has been expanded to include Tizen (an operating system by Samsung for proprietary devices). Additionally, Microsoft is working on supporting macOS, WPF, and GTK#. As you can imagine, this opens incredible opportunities in the market of cross-platform development because you can target both mobile and desktop systems. In this book, I will target the most popular operating systems (Android, iOS, and Windows 10) because support for other platforms is not yet released (apart from Tizen). You can read the official documentation about targeting macOS, Tizen, WPF, and GTK# in your Xamarin.Forms projects, with the assumption that your shared code will not change.

Setting up the development environment

In order to build native mobile apps with Xamarin.Forms, you need Windows 10 as your operating system, and Microsoft Visual Studio 2019 as your development environment. You can download and install the Visual Studio 2019 Community edition for free and get all the necessary tools for Xamarin development. I will discuss the latest stable release of Xamarin.Forms, version 4.8, in this book, so make sure you install version 16.7.2 or later of Visual Studio 2019.

When you start the installation, you will need to select the **Mobile development with .NET** workload in the Visual Studio Installer (see Figure 1).



Modifying — Visual Studio Enterprise 2019 — 16.7.2

Desktop & Mobile (5)

☑ .NET desktop development
Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET Core and .NET...

□ Desktop development with C++
Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.

☑ Universal Windows Platform development
Create applications for the Universal Windows Platform with C#, VB, or optionally C++.

☑ Mobile development with .NET
Build cross-platform applications for iOS, Android or Windows using Xamarin.

□ Mobile development with C++
Build cross-platform applications for iOS, Android or Windows using C++.

Gaming (2)

□ Game development with Unity
Create 2D and 3D games with Unity, a powerful cross-platform development environment.

□ Game development with C++
Use the full power of C++ to build professional games powered by DirectX, Unreal, or Cocos2d.

Installation details

- › Visual Studio extension development
- › .NET Core cross-platform development
- ⌄ Individual components
  - ☑ NuGet package manager
  - ☑ .NET Framework 4.6.1 targeting pack
  - ☑ C# and Visual Basic Roslyn compilers
  - ☑ C# and Visual Basic
  - ☑ F# language support
  - ☑ Container development tools
  - ☑ .NET Framework 4.7.2 targeting pack
  - ☑ JavaScript and TypeScript language support
  - ☑ JavaScript diagnostics
  - ☑ MSBuild
  - ☑ Text Template Transformation
  - ☑ Razor Language Services
  - ☑ IIS Express
  - ☑ SQL ADAL runtime
  - ☑ SQL Server Express 2016 LocalDB
  - ☑ Connectivity and publishing tools
  - ☑ CLR data types for SQL Server
  - ☑ SQL Server ODBC Driver

Location
C:\Program Files (x86)\Microsoft Visual Studio\2019\Enterprise

Total space required 0 KB

By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.

Figure 1: Installing Xamarin development tools

When you select this workload, the Visual Studio Installer will download and install all the necessary tools to build apps for Android, iOS, and Windows. iOS requires additional configuration, described in the next section.

For Windows 10 development, you need additional tools and SDKs, which you can get by also selecting the **Universal Windows Platform development** workload. If you select the **Individual components** tab, you will have an option to check if Android and Windows emulators have been selected or to make a choice manually (see Figure 2).