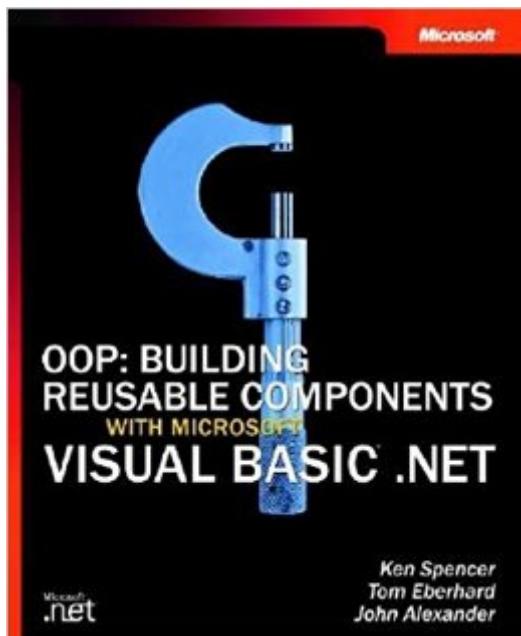# Oop: Building Reusable Components With Microsoft Visual Basic .Net

Synopsis

Aimed at experienced developers, this primer on writing Visual Basic .NET components explains how to design and build applications from reusable, shared components and how to construct an object-oriented architecture that simplifies the creation of complex business applications from reusable compone

Book Information

Paperback: 528 pages

Publisher: Microsoft Press (November 9, 2002)

Language: English

ISBN-10: 0735613796

ISBN-13: 978-0735613799

Product Dimensions: 7.1 x 1.3 x 9.4 inches

Shipping Weight: 2.2 pounds

Average Customer Review: 2.9 out of 5 starsÂ Â See all reviewsÂ (9 customer reviews)

Best Sellers Rank: #5,899,120 in Books (See Top 100 in Books)   #47 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Reuse   #1417 inÂ Books > Computers & Technology > Programming > Languages & Tools > Visual Basic   #1861 inÂ Books > Computers & Technology > Programming > Microsoft Programming > .NET

Customer Reviews

The .Net framework is huge. After professionally developing several web projects I had the nuts and bolts pretty much nailed. However, putting it all together into a sound model/approach is tough.I wish I'd read this book earlier on because it would have saved me a lot of pain. After you're past the novice stage and have a handle on the VB.Net syntax and object model, this book is the next step. It contains lots of code snippets and you can download their entire code library used to build the sample apps but the real key here is learning a sound methology.One review was critical of their approach. I disagree with his comments. This book offers not only a sound approach to application design using .Net (with some concentration on web development), n-tier architecture is accepted practice. The book offers a sound VB.Net implementation. Of course, it's hardly the only way and is a tad simplistic for the real world intranet apps I'm working on. But it should help intermediate programmers put it all together to move to the next level.My only complaint is that I wish it contained more code details, some broader coverage, and was more advanced. But that's a personal gripe because by the time I'd picked this up I'd personally grown past its content through the school of

hard knocks.

I have only read 9 books related to vb.net and most of them seem to be a copy of the MSDN library. This book is underrated but I found this book to be precise what I needed. It gives real world solutions, exploiting many of the .NET framework classes. While in other books only two or three chapters are of value this book is withinh the TOP on my list. This book is different because you are creating a full enterprise application, building different classes with specific purposes and then putting it together to have an End-product. It is this putting together that will help you understand the potentials of VB.NET

First of all, this book has almost nothing to do with OOP. It has some useful tidbits, such as using custom (server) controls to reduce coding and to increase standardization.However, I cannot recommend this title because of the many poor practices and bad application design it advocates. Why Microsoft Press doesn't insist it's title have some consistency in the use of naming standards, coding standards, application design, etc., is beyond me.The application design is poor or maybe the examples are just poorly done. The business layer doesn't appear to enforce any business rules while the data-access layer is also unconventially designed.Only one-third of this book (regarding server controls and UI design) is of any use. The rest is just junk.

Well, if you have absolutely NO experience with OOP and were looking for a good book to help you learn it . . . don't get this. :PI think most of the practices in this book are poor. For example, and this is simply personaly opinion, there is a section on using a data access layer that they wrote (I think it stinks). Hardly a very good DAL in my opinion. The thing I don't like about it is when they tell you to use it from other layers, you still have to know about this concept of SQLServer, connection strings, stored procedures, and parameters, and that is just lame. When I write a DAL, I abstract that stuff out. How in the world would I cleanly swap my DAL out if I wanted to go from SQL Server to Oracle?! Well, using their techniques it seems that I would have to go through the code in my other layers and change quite a bit . . . Horrible!Then, when they talk about using a security layer, they say this: "Our first thought was to create the security layer and have ASP.NET developers simply place code to check the login status at the top of each page. This approach is fraught with problems because it places the responsibility on the developer to work out the security on each page and include it." So, what do they do 2 paragraphs later? Tell you to put this header control that will handle it into EVERY page: "All the developer needs to do is put the header on the page and the

page is secure." Big deal, they abstract the security into one component, and then call that from this header. But the developer still needs to stick it in every page . . . (I personally would have used inheritence for this, but whatever works.)All in all, I think this book is about 5% useful. Seems like a good article would have accomplished the same goal, I don't see that this book has much valuable information. Thankfully I didn't pay for it, just read it via O'Reilly's Safari Subscription service!

This book is now required by my development staff to read before developing .NET applications. The first is "Practical Standards for Microsoft Visual Basic .NET". This book picks up where the other has left off.There are hundreds of ways to develop software, but Spencer, Eberhard and Alexander have done a great job illustrating one approach that is very practical for an enterprise enviroment. This book shows goes over many of the steps in software development and how to design the many tiers using .NET inheritance and reusing of code.My company name is Wharton Computer Consulting and we have been developing software for nearly a decade. I have read over a hundred technical books and have over 20 Microsoft Certifications. This book will allow me to spend less time going over how code should be done. During my technical reviews I will be able to point to chapters in this book on how it should be done.I have also taught at several university's and this will now be my recommended book for students interested in learning how to develop using .NET.

Oop: Building Reusable Components with Microsoft Visual Basic .Net MCAD/MCSD Self-Paced Training Kit: Developing WindowsÃ Â®-Based Applications with MicrosoftÃ Â® Visual BasicÃ Â® .NET and Microsoft Visual C#Ã Â® .NET, Second Ed: ... C#(r) .Net, Second Ed (Pro-Certification) Visual Developer Developing Custom Delphi 3 Components: Master the Art of Creating Powerful Delphi 3 Software Components Expert Oracle Application Express Plugins: Building Reusable Components (Expert's Voice in Oracle) MCPD Self-Paced Training Kit (Exams 70-536, 70-528, 70-547): MicrosoftÃ Â® .NET Framework Web Developer Core Requirements: Microsoft .Net Framework Web ... Requirements (Microsoft Press Training Kit) Database Programming with Visual Basic .NET and ADO.NET: Tips, Tutorials, and Code Developing Custom Delphi Components: Master the Art of Creating Powerful Delphi Software Components Software Engineering with Reusable Components CommonKADS Library for Expertise Modelling, Reusable Problem Solving Components (Frontiers in Artificial Intelligence and Applications) Apache Jakarta Commons: Reusable Java(TM) Components Reusable Ada Components Sourcebook (The Ada Companion Series) Office 2016 For Beginners- The PERFECT Guide on Microsoft Office: Including Microsoft

Excel Microsoft PowerPoint Microsoft Word Microsoft Access and more! Programming Distributed Applications with Com and Microsoft Visual Basic 6.0 (Programming/Visual Basic) Microsoft(r) Visual C++(r) .Net Step by Step--Version 2003 (Step by Step (Microsoft)) Microsoft Visual C++ .Net Step by Step (Step by Step (Microsoft)) Programming with Microsoft Visual Basic 2010 (VB.Net Programming) Dan Appleman's Developing COM/ActiveX Components With Visual Basic 6 Dan Appleman's Developing Activex Components With Visual Basic 5.0: A Guide to the Perplexed Pro Microsoft Speech Server 2007: Developing Speech Enabled Applications with .NET (Expert's Voice in .NET) Building Web Solutions with ASP.Net and ADO.NET (Developer Reference)

Dmca