

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code

By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek



Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek

Have you ever felt frustrated working with someone else's code? Difficult-to-maintain source code is a big problem in software development today, leading to costly delays and defects. Be part of the solution. With this practical book, you'll learn 10 easy-to-follow guidelines for delivering C# software that's easy to maintain and adapt. These guidelines have been derived from analyzing hundreds of real-world systems.

Written by consultants from the Software Improvement Group (SIG), this book provides clear and concise explanations, with advice for turning the guidelines into practice. Examples for this edition are written in C#, while our companion Java book provides clear examples in that language.

- Write short units of code: limit the length of methods and constructors
- Write simple units of code: limit the number of branch points per method
- Write code once, rather than risk copying buggy code
- Keep unit interfaces small by extracting parameters into objects
- Separate concerns to avoid building large classes
- Couple architecture components loosely
- Balance the number and size of top-level components in your code
- Keep your codebase as small as possible
- Automate tests for your codebase
- Write clean code, avoiding "code smells" that indicate deeper problems

 [Download Building Maintainable Software, C# Edition: Ten Gu ...pdf](#)

 [Read Online Building Maintainable Software, C# Edition: Ten ...pdf](#)

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code

By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek

Have you ever felt frustrated working with someone else's code? Difficult-to-maintain source code is a big problem in software development today, leading to costly delays and defects. Be part of the solution. With this practical book, you'll learn 10 easy-to-follow guidelines for delivering C# software that's easy to maintain and adapt. These guidelines have been derived from analyzing hundreds of real-world systems.

Written by consultants from the Software Improvement Group (SIG), this book provides clear and concise explanations, with advice for turning the guidelines into practice. Examples for this edition are written in C#, while our companion Java book provides clear examples in that language.

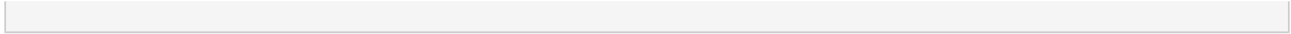
- Write short units of code: limit the length of methods and constructors
- Write simple units of code: limit the number of branch points per method
- Write code once, rather than risk copying buggy code
- Keep unit interfaces small by extracting parameters into objects
- Separate concerns to avoid building large classes
- Couple architecture components loosely
- Balance the number and size of top-level components in your code
- Keep your codebase as small as possible
- Automate tests for your codebase
- Write clean code, avoiding "code smells" that indicate deeper problems

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek **Bibliography**

- Rank: #780692 in Books
- Brand: O Reilly Media
- Published on: 2016-06-23
- Original language: English
- Dimensions: 9.17" h x .37" w x 7.01" l, .63 pounds
- Binding: Paperback
- 172 pages

 [Download Building Maintainable Software, C# Edition: Ten Gu ...pdf](#)

 [Read Online Building Maintainable Software, C# Edition: Ten ...pdf](#)



Download and Read Free Online Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek

Editorial Review

About the Author

Joost Visser is Head of Research at the Software Improvement Group. In this role, he is responsible for the science behind the methods and tools that SIG offers to measure and master software. Joost also holds a position as professor of Large Scale Software Systems at Radboud University Nijmegen. He has obtained his PhD in Computer Science from the University of Amsterdam and has published over 100 papers on topics such as generic programming, program transformation, green computing, software quality, and software evolution. Joost considers software engineering as a sociotechnical discipline and he is convinced that software measurement is essential for development teams and product owners to thrive.

Sylvan Rigal works as a software quality consultant at SIG since 2011 and is advising clients on managing their IT since 2008. He helps clients achieve lower software maintenance costs and enhanced security by prioritizing improvements in software design and development processes. He holds a MSc in international business from Maastricht University, The Netherlands (2006). As an active member of SIG's software security team, Sylvan trains consultants on analyzing software security risks. When he is not assessing technical health of software, he is training Brazilian jiu jitsu, enjoying Amsterdam's restaurants or traveling Asia.

Gijs Wijnholds joined the Software Improvement Group in 2015 as a software quality consultant in public administration. He helps clients get in control of their software projects by advising them on development processes and translating technical risks into strategic decisions. Gijs holds a BSc in AI from Utrecht University and a MSc degree in Logic from University of Amsterdam. He is an expert on Haskell and mathematical linguistics.

Pascal van Eck joined the Software Improvement Group (SIG) in 2013 as a general consultant on software quality. Prior to joining SIG, for 13 years Pascal was Assistant Professor of Information Systems at University of Twente, The Netherlands. Pascal holds a PhD in Computer Science from Vrije Universiteit Amsterdam and has published over 80 papers in areas such as enterprise architecture, IT security, and software metrics. Pascal is chairman of the program committee of the Dutch National Conference on Architecture for The Digital World.

After obtaining an MSc degree in Software Engineering from Delft University of Technology in 2005, Rob joined SIG as a software quality consultant. Working at SIG is for Rob the closest thing to being a software doctor. In his role as a consultant he combines his thorough technical knowledge on software engineering and software technologies to advice clients how to keep their systems in shape. Next to being a consultant, Rob fulfills a leading role in SIG's internal development team. This team develops and maintains the company's software analysis tooling. It's Rob's ambition to leave the IT industry a bit better than he found it.

Users Review

From reader reviews:

John Loya:

The knowledge that you get from Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code may be the more deep you excavating the information that hide within the words the more you get interested in reading it. It does not mean that this book is hard to recognise but Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code giving you thrill feeling of reading. The article author conveys their point in specific way that can be understood by anyone who read the item because the author of this guide is well-known enough. This book also makes your own vocabulary increase well. Making it easy to understand then can go along, both in printed or e-book style are available. We suggest you for having this Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code instantly.

Karen Partain:

Reading a reserve tends to be new life style in this era globalization. With reading through you can get a lot of information that may give you benefit in your life. Along with book everyone in this world can easily share their idea. Textbooks can also inspire a lot of people. Many author can inspire their reader with their story or their experience. Not only the storyline that share in the publications. But also they write about the data about something that you need illustration. How to get the good score toefl, or how to teach children, there are many kinds of book that you can get now. The authors nowadays always try to improve their proficiency in writing, they also doing some investigation before they write for their book. One of them is this Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code.

Adrian Kester:

Do you have something that you want such as book? The book lovers usually prefer to pick book like comic, quick story and the biggest some may be novel. Now, why not striving Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code that give your pleasure preference will be satisfied simply by reading this book. Reading addiction all over the world can be said as the opportunity for people to know world far better then how they react when it comes to the world. It can't be said constantly that reading practice only for the geeky person but for all of you who wants to become success person. So , for every you who want to start reading through as your good habit, you may pick Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code become your current starter.

Janelle Coe:

Do you like reading a guide? Confuse to looking for your preferred book? Or your book had been rare? Why so many problem for the book? But just about any people feel that they enjoy for reading. Some people likes studying, not only science book and also novel and Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code or maybe others sources were given know-how for you. After you know how the truly great a book, you feel wish to read more and more. Science publication was created for teacher or students especially. Those ebooks are helping them to put their knowledge. In various other case, beside

science reserve, any other book likes Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code to make your spare time much more colorful. Many types of book like this.

Download and Read Online Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek #5EJ8YC6BUR0

Read Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek for online ebook

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek books to read online.

Online Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek ebook PDF download

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek Doc

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek Mobipocket

Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code By Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek EPub