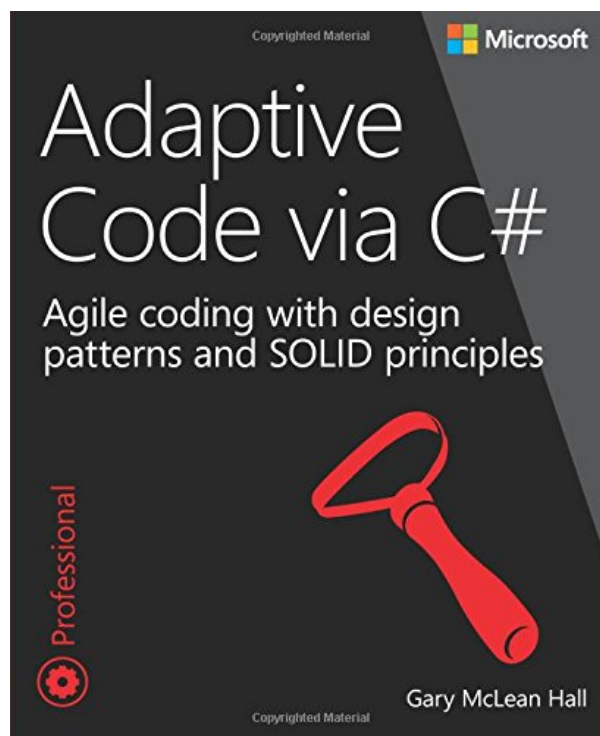
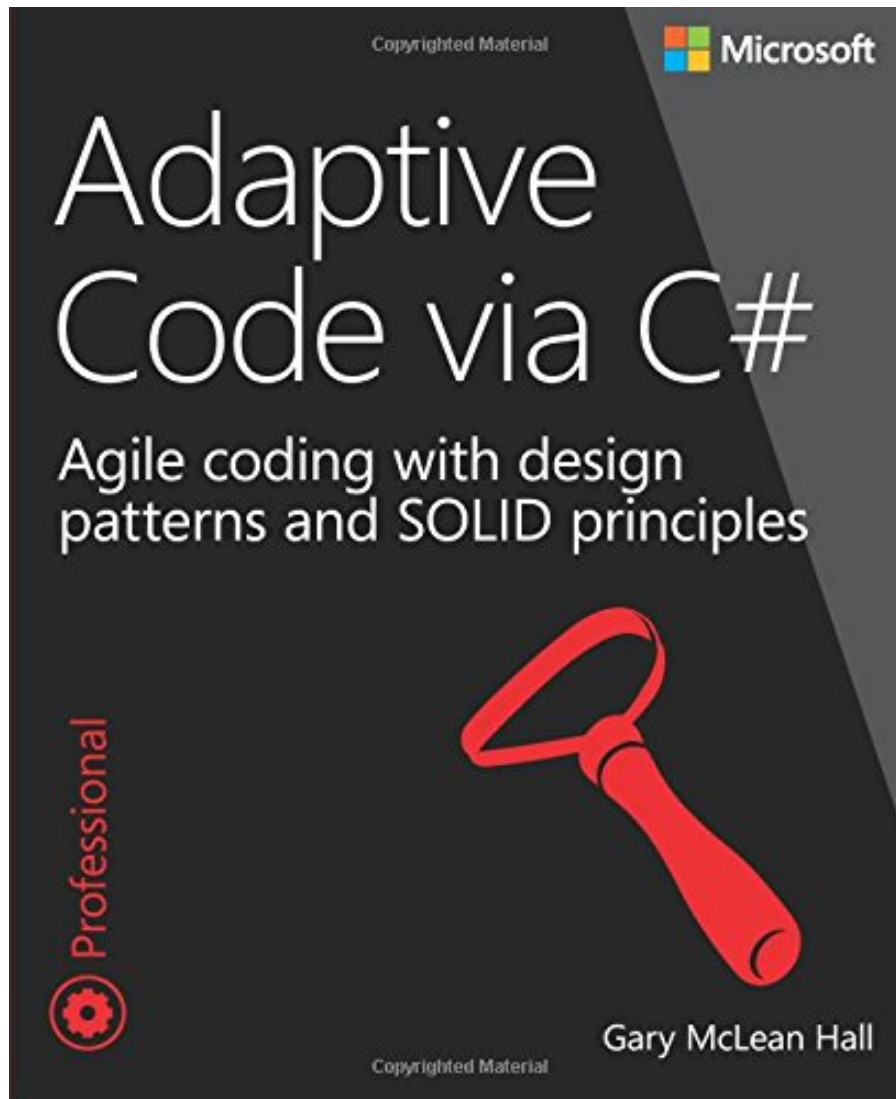


**ADAPTIVE CODE VIA C#: AGILE CODING  
WITH DESIGN PATTERNS AND SOLID  
PRINCIPLES (DEVELOPER REFERENCE) BY  
GARY MCLEAN HALL**



**DOWNLOAD EBOOK : ADAPTIVE CODE VIA C#: AGILE CODING WITH  
DESIGN PATTERNS AND SOLID PRINCIPLES (DEVELOPER REFERENCE) BY  
GARY MCLEAN HALL PDF**





Click link bellow and free register to download ebook:

**ADAPTIVE CODE VIA C#: AGILE CODING WITH DESIGN PATTERNS AND SOLID PRINCIPLES (DEVELOPER REFERENCE) BY GARY MCLEAN HALL**

[DOWNLOAD FROM OUR ONLINE LIBRARY](#)

# **ADAPTIVE CODE VIA C#: AGILE CODING WITH DESIGN PATTERNS AND SOLID PRINCIPLES (DEVELOPER REFERENCE) BY GARY MCLEAN HALL PDF**

This book *Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference)* By Gary McLean Hall offers you better of life that could create the top quality of the life more vibrant. This Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall is what the people now need. You are here as well as you might be exact and also certain to get this publication Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall Never ever question to get it even this is merely a publication. You could get this book Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall as one of your collections. But, not the compilation to show in your bookshelves. This is a valuable publication to be checking out compilation.

## About the Author

Gary McLean Hall specializes in patterns and practices, with particular focus on creating code that is adaptive to change.

# **ADAPTIVE CODE VIA C#: AGILE CODING WITH DESIGN PATTERNS AND SOLID PRINCIPLES (DEVELOPER REFERENCE) BY GARY MCLEAN HALL PDF**

[Download: ADAPTIVE CODE VIA C#: AGILE CODING WITH DESIGN PATTERNS AND SOLID PRINCIPLES \(DEVELOPER REFERENCE\) BY GARY MCLEAN HALL PDF](#)

Reviewing an e-book **Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall** is sort of simple activity to do every time you want. Also checking out every time you want, this task will certainly not interrupt your other tasks; many individuals frequently check out the books Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall when they are having the downtime. Just what regarding you? Just what do you do when having the downtime? Don't you spend for worthless things? This is why you should get the publication Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall and attempt to have reading routine. Reading this e-book Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall will not make you pointless. It will give a lot more advantages.

When going to take the experience or ideas forms others, publication *Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall* can be an excellent source. It holds true. You could read this Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall as the resource that can be downloaded and install here. The means to download and install is also easy. You could see the link page that we offer and afterwards buy the book making a deal. Download Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall and also you could put aside in your personal tool.

Downloading guide Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall in this web site listings can make you more advantages. It will certainly reveal you the very best book collections and also finished compilations. So many publications can be discovered in this internet site. So, this is not just this Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall Nonetheless, this book is referred to review because it is a motivating publication to give you a lot more possibility to get encounters and thoughts. This is easy, check out the soft documents of guide [Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles \(Developer Reference\) By Gary McLean Hall](#) and you get it.

# **ADAPTIVE CODE VIA C#: AGILE CODING WITH DESIGN PATTERNS AND SOLID PRINCIPLES (DEVELOPER REFERENCE) BY GARY MCLEAN HALL PDF**

Agile coding with design patterns and SOLID principles

As every developer knows, requirements are subject to change. But when you build adaptability into your code, you can respond to change more easily and avoid disruptive rework. Focusing on Agile programming, this book describes the best practices, principles, and patterns that enable you to create flexible, adaptive code--and deliver better business value.

Expert guidance to bridge the gap between theory and practice

- Get grounded in Scrum: artifacts, roles, metrics, phases
- Organize and manage architectural dependencies
- Review best practices for patterns and anti-patterns
- Master SOLID principles: single-responsibility, open/closed, Liskov substitution
- Manage the versatility of interfaces for adaptive code
- Perform unit testing and refactoring in tandem
- See how delegation and abstraction impact code adaptability
- Learn best ways to implement dependency interjection
- Apply what you learn to a pragmatic, agile coding project

Get code samples at:

<http://github.com/garymclean/AdaptiveCode>

- Sales Rank: #102730 in Books
- Published on: 2014-10-19
- Released on: 2014-10-09
- Original language: English
- Number of items: 1
- Dimensions: 8.90" h x 1.10" w x 7.40" l, .0 pounds
- Binding: Paperback
- 448 pages

About the Author

Gary McLean Hall specializes in patterns and practices, with particular focus on creating code that is adaptive to change.

Most helpful customer reviews

28 of 31 people found the following review helpful.

Ready to get to the next level?

By William Orick

The description of the book is pretty good, so let me tell you what I think of the book instead of what it covers. Let me start with an analogy. Let's say you are learning to be a carpenter. You just found out how to use a hammer, screw driver, tape measure, and a few saws. Think you can build a strong house to code and regulations now? No, not even close. Just because you know how to use tools, doesn't mean you know WHEN to use them. Should I use screws, or nails for that? How many nails? what size? This part is what separates someone with knowledge of programming from a great programmer. Maybe you took some classes, or even got a degree. Great, now you have a tool bag full of programming tools and you mostly know what they do. This book will teach you the patterns and habits to follow to best use those tools the right way. If all you want are the 1, 2, 3 steps to follow, you can get that quickly. If you want to know why these are the best steps, the book offers that up as well.

The book suggests you should have at least have a firm grasp on basic programming skills to get the most out of it. I do agree that a good part of the book would be beyond someone who doesn't know what a variable is, but I think this book should be on every programmers bookshelf regardless of his or her skill level. For a beginner programmer this book lays a good foundation on programming habits and ways of thinking that will really help you become a skilled professional. The book really shines for someone who took some classes and has worked as a developer for a year or two. At this point you have the syntax down on at least one or two languages and are realizing that was just the first step. Even advanced programmers will pick up several good idea's.

While the book is full of great information, I think my favorite part are the last two chapters that follow a fictitious developer group working on projects. They go step by step through the processes. The book gives considerable attention and space to dialogs from the team discussing the projects between themselves. While the wording sounds a bit scripted at times, it does a really good job of highlighting how a development group interacts, and what kind of issues and concerns often come up. I think that part really helps achieve the books goal of bridging theory into real world application. The skills and knowledge you get from this book are central to those seemingly nebulous soft skills that win people jobs and promotions. If I have ever seen experience in a bottle, this is it.

7 of 8 people found the following review helpful.

Great Book On Modern Software Design

By Matthew Coleman

TL;DR - This book is a great read for all levels of software engineers. It is a good reminder and description of well accept software design principles. It provides practical examples of how these principles can be applied to real-world software products. Personally, I wrote notes of a number of items to integrate into my team's process. I would recommend this book for anyone who has a good grasp on the C# language.

I am a development manager with 15 years of experience and in charge of a large system that is in need of a good refactoring. I was interested in this book to help update the teams design guidelines. I was impressed with the content of this book. This is the first programming book that has impressed me in the last few years.

This book has two main focuses the SCRUM process and the SOLID principles. It is divided into three parts: An Agile Foundation, Writing Solid Code, and a sample application.

The first part, An Agile Foundation, focuses on the SCRUM process. In my experience, I have never worked in an organization that would fully support a SCRUM development process. However, only one chapter in this part of the book is fully dedicated to the description of the SCRUM process. The remaining chapters bring up many topics such as technical debt, refactoring, unit testing, design patterns. All of the topics

discussed should be part of any technical teams design process.

The second part, Writing SOLID Code, focus on the SOLID design:

Single Responsibility Principle

Open/Close Principle

Liskov Substitution Principle

Interface Segregation

Dependency Injection

The SOLID principles have been the focus of numerous blog posts and podcasts over the past year. All of these avenues have provided a great summary of the principles. However, I have not found any site that provide details of each principle and provides examples. In addition, the author also gives examples where using one or a combination of these principles are overkill. I think this is missing in most of the descriptions I have previously seen.

The last section of the book was a dedicated example combining the SCRUM process with the SOLID design principles. I thought this was a really useful section for all levels of development. For an more experience developer it gave examples of how to integrate these principles in their current design process. For the inexperience developer this gave a great example of the complete process.

I would suggest that the author consider examples with more traditional process (not quite waterfall, not quite agile). This way it would quickly be applicable to the vast majority of developers.

9 of 11 people found the following review helpful.

Please DO read this book!

By Markus Schmidt

This book is a game-changer for the C# community. Scrum, patterns, the SOLID principles and clean code all have their roots more or less in the Java language. Development on the Microsoft stack was dominated a long time by an extremely pragmatic and tool-centric style. Fortunately, in the last years more and more enterprise developers are starting to build large systems on the .NET Framework, not few of them being former Java developers, creating the need for more flexible architectures and a source code structure which does not start rotting on day one.

The book has the following structure:

## PART I AN AGILE FOUNDATION

CHAPTER 1 Introduction to Scrum

CHAPTER 2 Dependencies and layering

CHAPTER 3 Interfaces and design patterns

CHAPTER 4 Unit testing and refactoring

## PART II WRITING SOLID CODE

CHAPTER 5 The single responsibility principle

CHAPTER 6 The open/closed principle

CHAPTER 7 The Liskov substitution principle

CHAPTER 8 Interface segregation

CHAPTER 9 Dependency injection

## PART III ADAPTIVE SAMPLE

CHAPTER 10 Adaptive sample: Introduction

CHAPTER 11 Adaptive sample: Sprint 1

CHAPTER 12 Adaptive sample: Sprint 2

Part 1 defines the building blocks of professional software development. After a short introduction into scrum, the main problem of maintainable software is discussed from different viewing angles: dependencies. The ubiquitous tool to manage dependencies is the topic of the next chapter: No pattern without an interface. Unit tests and refactorings are the last chapter of this part and it emphasizes the important fact that unit tests are the prerequisite for refactoring and code that is not permanently adopted starts rotting. I like the writing style of the author, as he does not talk too much about theory. Instead he makes examples and show the consequences of the different approaches. You can easily follow his arguments and its fun reading.

Part 2 iterates through the SOLID principles using many good code examples. He starts with the typical coding style of the intermediate programmer, which unfortunately reminded me of some code I wrote myself, gradually refactoring it towards clean code. This process is accompanied by exhaustive reflection of the resulting quality and also shows the limits of perfection. I like most that he takes into account, that good code is not definable theoretically, but depends on the team, the experience, the customer and the whole situation.

Part 3 lets the reader be a fly on the wall of a scrum team. It shows how the team deals with changing requirements and how the SOLID principles helps them succeeding. This part leaves no doubt that the main target of this book is learning by reading and reviewing code. The many examples are all downloadable from GitHub.

The author Gary McLean Hall crafted a book, which is more than the sum of its parts. It's a concise and at the same time very practical summary of the main agile topics of the last decade rewritten from a C# perspective. It's full of code examples which help programmers to grasp the ideas behind the principles and patterns. The biggest difference to other development books in this area is, that it is not dogmatic. The author leaves no doubt, that the shown coding techniques are very powerful, but he also concedes that sometimes a simple class violating SRP is appropriate.

Garry McLean Hall coined the term "adaptive code" and I am sure that this was not the last thing we heard of him.

See all 70 customer reviews...

# **ADAPTIVE CODE VIA C#: AGILE CODING WITH DESIGN PATTERNS AND SOLID PRINCIPLES (DEVELOPER REFERENCE) BY GARY MCLEAN HALL PDF**

Your impression of this publication **Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall** will certainly lead you to get exactly what you precisely require. As one of the impressive publications, this book will provide the visibility of this ledged Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall to collect. Also it is juts soft documents; it can be your collective file in gizmo and other gadget. The essential is that usage this soft data publication Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall to check out as well as take the benefits. It is just what we mean as publication Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall will certainly enhance your ideas as well as mind. After that, reading publication will likewise enhance your life quality better by taking great action in balanced.

## About the Author

Gary McLean Hall specializes in patterns and practices, with particular focus on creating code that is adaptive to change.

This book *Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall* offers you better of life that could create the top quality of the life more vibrant. This Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall is what the people now need. You are here as well as you might be exact and also certain to get this publication Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall Never ever question to get it even this is merely a publication. You could get this book Adaptive Code Via C#: Agile Coding With Design Patterns And SOLID Principles (Developer Reference) By Gary McLean Hall as one of your collections. But, not the compilation to show in your bookshelves. This is a valuable publication to be checking out compilation.