



Software Craftsmanship: The New Imperative

By *Pete McBreen*

Download now

Read Online 

Software Craftsmanship: The New Imperative By Pete McBreen

Introducing software craftsmanship, a programmer-centric way to build software, this text explains why software engineering isn't enough and aims to transform the developer's relationship with users and customers.

 [Download Software Craftsmanship: The New Imperative ...pdf](#)

 [Read Online Software Craftsmanship: The New Imperative ...pdf](#)

Software Craftsmanship: The New Imperative

By Pete McBreen

Software Craftsmanship: The New Imperative By Pete McBreen

Introducing software craftsmanship, a programmer-centric way to build software, this text explains why software engineering isn't enough and aims to transform the developer's relationship with users and customers.

Software Craftsmanship: The New Imperative By Pete McBreen Bibliography

- Sales Rank: #443121 in Books
- Published on: 2001-09-02
- Released on: 2001-08-23
- Original language: English
- Number of items: 1
- Dimensions: 9.00" h x .60" w x 7.40" l, .95 pounds
- Binding: Paperback
- 208 pages

 [Download Software Craftsmanship: The New Imperative ...pdf](#)

 [Read Online Software Craftsmanship: The New Imperative ...pdf](#)

Editorial Review

From the Back Cover

By recognizing that software development is not a mechanical task, you can create better applications.

Today's software development projects are often based on the traditional software engineering model, which was created to develop large-scale defense projects. Projects that use this antiquated industrial model tend to take longer, promise more, and deliver less.

As the demand for software has exploded, the software engineering establishment has attempted to adapt to the changing times with short training programs that teach the syntax of coding languages. But writing code is no longer the hard part of development; the hard part is figuring out what to write. This kind of know-how demands a skilled craftsman, not someone who knows only how to pass a certification course.

Software Craftsmanship presents an alternative—a craft model that focuses on the people involved in commercial software development. This book illustrates that it is imperative to turn from the technology-for-its-own-sake model to one that is grounded in delivering value to customers. The author, **Pete McBreen**, presents a method to nurture mastery in the programmer, develop creative collaboration in small developer teams, and enhance communications with the customer. The end result—skilled developers who can create, extend, and enhance robust applications.

This book addresses the following topics, among others:

Managing software craftsmen *Software Craftsmanship* is written for programmers who want to become exceptional at their craft and for the project manager who wants to hire them.

0201733862B07242001 About the Author

Pete McBreen is an independent consultant who actually enjoys writing and delivering software. Despite spending a lot of time writing, teaching, and mentoring, he goes out of his way to ensure that he does hands-on coding on a live project every year. Pete specializes in finding creative solutions to the problems that software developers face. After many years of working on formal and informal process improvement initiatives, he took a sideways look at the problem and realized, “Software development is meant to be fun. If it isn't, the process is wrong.” Pete lives in Cochrane, Alberta, Canada and has no plans to move back to a big city.

0201733862AB07092002 Excerpt. © Reprinted by permission. All rights reserved.

Craftsmanship is a return to the roots of software development: Good software developers have always understood that programming is a craft skill. Regardless of the amount of arcane and detailed technical knowledge that a person has, in the end, application development comes down to feel and experience. Someone can know all of the esoteric technical details of the Java programming language, but that person will never be able to master application development unless he or she develops a feel for the aesthetics of software. Conversely, once a person gets the feel for software development, the specific technical details become almost irrelevant. Great developers are always picking up and using new technology and techniques; learning a new technology is just a normal part of the life of a software developer. The term software engineering was coined in 1967 by a NATO study group that recommended a conference to discuss “the problems of software.” The report from this 1968 conference, which was sponsored by the NATO Science

Committee and took place in Garmish, Germany, was titled Software Engineering.¹ In the report, Peter Naur and Brian Randell stated, “The phrase ‘software engineering’ was deliberately chosen to be provocative, in implying the need for software manufacture to be based on the types of theoretical foundations and practical disciplines that are traditional in the established branches of engineering.” In the same spirit, it is the intention of this book to be deliberately provocative in implying the need for practitioners to start paying attention to the craft of software development. Software craftsmanship is important because it takes us away from the manufacturing metaphor that software engineering invokes and makes us pay attention to the people who do software development. Craftsmanship brings with it the metaphor of skilled practitioners intent on mastering their craft, of pride in and responsibility for, the fruits of their labor. Software craftsmanship is not the opposite of software engineering or computer science. Rather, craftsmanship is a different tradition that happily coexists with and benefits from science and engineering. Just as the modern blacksmith benefits from better tools, materials, and understanding, so software craftsmanship benefits from better computers, reusable components, and programming languages. Just as blacksmiths transcend science and engineering with their skill and artistry, software craftsmanship can transcend computer science and software engineering to produce great programs, applications, and systems. UNIX and the modern-day GNU Linux are probably the best-known examples of this—systems that are thriving due to the craft, skill, and dedication of their creators. Software craftsmanship is a response to the problems of trying to force-fit software engineering into commercial application development. Software engineering was developed to meet the needs of NATO in developing very large defense systems. Commercial application development differs from the development of defense and government systems in that applications are a whole lot smaller and normally have to be up and running in less than 18 months. It is rare for a commercial application to be developed by a team of more than 20 people, and most application developers work in teams with fewer than 10 members. Software engineering is good at handling the problems of really large teams of 200 or more people, but it has little to say about how the individuals in a team should practice their craft. Software engineering encourages the “human wave”² approach to software development. Rather than solving the problem of how to develop highly skilled developers, software engineering attempts to deskill software development by suggesting that every problem can be solved by throwing more people at it. Although this approach sometimes succeeds, the resulting software is junk. Slow and bloated, it just never feels right. Users are dazzled by the graphics and animation but never really manage to come to grips with the software. They are thwarted by their inability to learn the software and use only a small fraction of the available features. Software does not have to be like that. All too often I see application development teams shipping valuable applications that provide real, measurable business benefit, but apologizing for not following software engineering best practices. For me, the real test of a team is whether it manages to ship and then enhance and extend the application for years afterward. Timely shipping of the first release is important, but it is more important that subsequent releases occur in a timely fashion and that each new release improves the application. Whenever I’m asked about hiring developers, I tell people to look for developers who have shipped a few applications successfully and then stuck around long enough to handle the next enhancement or maintenance release. Shipping proves that the developer can make something work; staying around for the next release allows the developer to experience the effects of the way that he or she built the application in the first place. If a developer has done this three times, my guess is that he or she is skilled and experienced enough in the craft of software development to be successful again. Software craftsmanship is the new imperative because many members of the software development community are starting to chase technology for its own sake, forgetting what is important. The purpose of software development is to create high-quality, robust software applications that deliver value to their users. What matters is growing a new generation of developers who can do that. Software craftsmanship stands for putting the joy and excitement back into creating applications for our users.

¹ Naur, Peter, and Brian Randell, (eds.), *Software Engineering: A Report on a Conference Sponsored by the NATO Science Committee*, NATO, 1969.

2 Levy, Steven, Hackers, Penguin Books, 1994, p. 88. 0201733862P08202001 Users Review**From reader reviews:**

Richard Linneman:Software Craftmanship: The New Imperative can be one of your beginning books that are good idea. We recommend that straight away because this reserve has good vocabulary that can increase your knowledge in language, easy to understand, bit entertaining but nevertheless delivering the information. The writer giving his/her effort to get every word into joy arrangement in writing Software Craftmanship: The New Imperative nevertheless doesn't forget the main position, giving the reader the hottest along with based confirm resource details that maybe you can be one of it. This great information can certainly drawn you into brand new stage of crucial thinking.

Charles Thomas:That publication can make you to feel relax. This book Software Craftmanship: The New Imperative was colorful and of course has pictures around. As we know that book Software Craftmanship: The New Imperative has many kinds or variety. Start from kids until youngsters. For example Naruto or Investigation company Conan you can read and believe you are the character on there. So , not at all of book are usually make you bored, any it can make you feel happy, fun and unwind. Try to choose the best book in your case and try to like reading which.

Matthew German:A lot of publication has printed but it is unique. You can get it by world wide web on social media. You can choose the very best book for you, science, witty, novel, or whatever through searching from it. It is called of book Software Craftmanship: The New Imperative. Contain your knowledge by it. Without leaving the printed book, it can add your knowledge and make a person happier to read. It is most crucial that, you must aware about guide. It can bring you from one destination for a other place.

Omar Lamm:Many people said that they feel fed up when they reading a guide. They are directly felt the idea when they get a half portions of the book. You can choose the actual book Software Craftmanship: The New Imperative to make your own reading is interesting. Your own personal skill of reading skill is developing when you such as reading. Try to choose very simple book to make you enjoy to read it and mingle the feeling about book and studying especially. It is to be initial opinion for you to like to wide open a book and learn it. Beside that the e-book Software Craftmanship: The New Imperative can to be your friend when you're feel alone and confuse with what must you're doing of these time.

Download and Read Online Software Craftmanship: The New Imperative By Pete McBreen
#0QS6X29CJZA

Read Software Craftsmanship: The New Imperative By Pete McBreen for online ebook Software
Craftsmanship: The New Imperative By Pete McBreen Free PDF download, 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
Software Craftsmanship: The New Imperative By Pete McBreen books to read online. Online Software
Craftsmanship: The New Imperative By Pete McBreen ebook PDF download Software Craftsmanship: The
New Imperative By Pete McBreen Doc Software Craftsmanship: The New Imperative By Pete McBreen
Mobipocket Software Craftsmanship: The New Imperative By Pete McBreen EPub 0QS6X29CJZA: Software
Craftsmanship: The New Imperative By Pete McBreen