

## Foreword

After spending the summer north of the Arctic Circle, basking in the midnight sun and the warmest weather for over 100 years in Finland, I was especially happy to find this book sitting on my desk waiting to be read. Although there is no shortage of books on agile methodologies and practices, something had been missing. The concept of quality is indeed a very important element in any software system and development method, yet it has received little explicit attention in the agile literature. For this reason, I am delighted to see this book contribute to this gap.

We have long known that skilled people are the most crucial resource in software development. Back in the 1990 summer issue of *American Programmer* (Ed Yourdon's Software Journal, Vol. 3, No. 7-8)—which was devoted exclusively to “Peopleware”—the editor commented that “Everyone knows the best way to improve software productivity and quality is to focus on people.” However, it took more than 10 years for the agile manifesto and agile methods (Extreme Programming, Scrum, Crystal, and many others) to truly place the emphasis on people and their interaction. Since then, we have witnessed a movement that has advanced more rapidly than any other innovation in the field of software engineering.

Software quality in agile development is not a straightforward topic. Therefore, it is essential that a book of this kind does not aim at offering simple answers to complex problems. An edited book allows the contributors to approach the topic from their particular angles in an in-depth manner. In this book there are chapters not normally found in the agile literature dealing with, for example, metrics and documenting defects. Some of the chapters take a controversial approach and offer new insights into adapting agile methods in different development situations. The reader will quickly realise that these types of arguments, studies, and suggestions are much needed in this field.

The reader can further enjoy the opportunity to select and read the contents pertaining to their background and interests. I am happy to see that the editors have succeeded in collecting chapters that not only build upon one another but, more importantly, form a coherent whole addressing the relevant issues from people management to coding with experiences drawn from the industry. And all of this is addressed from the perspective of software quality!

As an academic, I value the fact that this book includes a number of rigorously performed scientific studies. This is particularly welcome as it enables us to answer the question why agile methods work. To date, we have seen quite interesting anecdotal evidence that agile methods do improve quality and even make the programmers' work a happier one. However, this book contributes also to the scientific discussion by providing thoughts and theories that explain the results.

Sometimes we tend to forget that one of the better ways to influence the future of software development is to offer specific material for teachers who educate young developers in universities and other

educational institutes. While I believe all the chapters are of merit in this book, I am impressed to find a chapter written for the use of educators as well.

Whether you read this book from start to finish, or piecemeal your approach iteratively, I am sure you will find this book as valuable as I did.

*Pekka Abrahamsson*

*Research Professor*

*VTT Technical Research Centre of Finland*

**Pekka Abrahamsson** is a research professor at VTT Technical Research Centre of Finland. Currently, he is on leave from the University of Tampere where he is a full professor in the field of information systems and software engineering. His current responsibilities include managing an AGILE-ITEA project (<http://www.agile-itea.org>), which involves 22 organizations from nine European countries. The project aims at developing agile innovations in the domain of complex embedded systems. His research interests are centred on mobile application development, business agility, agile software production, and embedded systems. He leads the team who has designed an agile approach for mobile application development—the Mobile-D. He has coached several agile software development projects in industry and authored 50+ scientific publications focusing on software process and quality improvement, agile software development and mobile software. His professional experience involves 5 years in industry as a software engineer and a quality manager