

《敏捷开发的艺术》

图书基本信息

书名：《敏捷开发的艺术》

13位ISBN编号：9787111268048

10位ISBN编号：7111268040

出版时间：2009-8

出版社：机械工业出版社

作者：James Shore,Shane Warden

页数：448

译者：王江平

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《敏捷开发的艺术》

内容概要

本书为那些正在考虑应用敏捷开发来构建有价值软件的人们提供了实用的指导。现在已经有大量的书籍描述敏捷开发是什么或者为什么它能帮助软件项目成功，但很少有哪一本书能把针对开发者、管理者、测试者和客户的信息合并成一个整体，从而使其能够直接应用。

本书为敏捷的计划、开发、交付和管理提供了严谨的建议，这些建议来自于作者多年的极限编程(Extreme Programming, XP)经验。你将看到敏捷开发过程的全景图，包括为非技术类读者准备的全面指导，以及为开发者和测试人员准备的实用技术实践。

本书为以下问题提供了明确的答案：

怎样才能采用敏捷开发？

我们真的需要结对编程吗？

汇报应该详细到什么程度？

如果无法让客户参与进来该怎么办？

我们应该编写多少文档？

何时进行设计和架构？

作为一名非开发人员，我应如何同敏捷团队一起工作？

产品的路线在哪里？

QA应该如何参与进来？

本书教你如何采用XP实践，详细描述了每一种实践，然后讨论了一些原则，使你可以更改XP并创建自己的敏捷方法。尤其是，本书为敏捷开发中一些较为困难的方面(合作的需要和团队成员之间的信任)提供了解决办法。

不管你目前已经是敏捷团队的一部分，还是只对敏捷开发感兴趣，本书都为你提供了开始实践敏捷开发所需的实用技巧。随着你的经验的增长，内容也随之深入。本书教你首先理解敏捷开发的规则，然后打破这些规则，最后当你掌握了敏捷开发的艺术之后，再完全撇开这些规则。

《敏捷开发的艺术》

书籍目录

前言第1部分 入门 第1章 为什么需要敏捷 理解成功 成功不只是如期完成 组织成功的重要性 走进敏捷 第2章 如何做到敏捷 敏捷方法 不要自己炮制方法 精通之道 寻找一位导师 第3章 理解XP XP生命周期 XP团队 XP概念 第4章 采用XP XP适合我们吗 第4章 采用XP XP适合我们吗 现在开始! 评估你的敏捷度第2部分 实践X 第5章 思考 结对编程 精力充沛地工作 信息化工作场所 根源分析 回顾 第6章 协作 信任 坐到一起 真实客户参与 统一协作语言 站立会议 编码规范 迭代演示 汇报 第7章 发布 全部完成 没有bug 版本控制 十分钟构建 持续集成 代码集体所有制 文档 第8章 计划 第9章 开发第3部分 掌握敏捷 第10章 价值和原则 第11章 改善过程 第12章 以人为本 第13章 消除浪费 第14章 交付价值 第15章 寻求技术卓越参考文献

《敏捷开发的艺术》

精彩短评

- 1、我读完了.....
- 2、写的很全面
- 3、迭代生命周期，开发思想进步的一本书。
- 4、南图借的,TP311.52/0193
- 5、讲的非常详细！
- 6、之前简单的翻看过一遍，有需要的时候可以拿出来再精读
- 7、continuous integration
Courage,Communication,Simplicity,Feedback,Respect.
- 8、关于XP的好书。
- 9、看完之后就明白为什么这么需要敏捷教练了,因为可以注意的细节和角度太多了.内容很充实,能凑这么厚一本书真不容易呀加一星
- 10、完全实战型，非理想主义者的敏捷开发指南。
- 11、先是英文版读了一半，买了中文版后继续读。信息量大，可以作为工具书放在身边，随时翻阅
- 12、会引起思考
- 13、：
TP311.52/1629
- 14、敏捷开发的执行手册，主要内容应该是偏XP，涵盖的面很全，初涉敏捷的团队可以用来参考。
- 15、关键字:结对编程/迭代式发布/舒服的双人大座位设计的工位...
- 16、这是我目前读到的关于敏捷开发，最全面、最详实、论证最充分的一本书。
- 17、关于敏捷开发的实践介绍，各种经验之谈，作为开发指导非常有用。
- 18、随手一翻
- 19、目前读过的XP的最好的书
- 20、经典啊，很详细。读后就知道怎么实践了，不过页数真的很长，看了好长时间.....
- 21、泛读了一下，对作者描述的工作场景很喜欢，mark一下，将来用到时再认真看看
- 22、是不是叫《敏捷开发最佳实践》会更好点，艺术就有点拔高了。其实是比较务实的书，就是有点儿乱，翻译得不好可能也是其中一个原因。
- 23、真正讲如何实施敏捷开发的书，可惜木有机会实践.....
- 24、介绍了很多XP的实践，比如PP, TDD。概念性的东西居多，适合迷惑的时候拿来温习一下。或者敏捷新兵阅读。
- 25、用了一天的时间翻完这本书，主要是大概的了解一下敏捷开发这种模式。其中的某些技术指导优点很明显，迭代式开发、TTD等。还有某些产品上的建议，再次科普了。
- 26、敏捷实践案头书之一。
- 27、大概了解一下敏捷。

- 1、在译者的豆瓣小站里发了勘误信息：<http://site.douban.com/widget/notes/3854525/note/182077542/>欢迎大家来提交发现的错误：<http://site.douban.com/120940/room/804624/>
- 2、春节期间断断续续把《敏捷开发的艺术》看完了，虽然看的虎头蛇尾，但也有收获。最初在微博上知道敏捷开发，一群技术牛人讲敏捷开发如何地好，并且一个InfoQ的组织在全国各地做敏捷开发讲座，带团队。最让人兴奋的是听说腾讯等一大批互联网前沿公司从02年左右就开始应用敏捷开发。仰慕腾讯惊人的开发速度与质量，所以就持续了解敏捷开发，又因工作所需，就在网上找了这本认为敏捷开发方面最好的书。书上对敏捷开发是这样定义的：一，结对工作；二，测试驱动开发；三，分析、设计、编码、测试这一切同时发生，小范围迭代，持续前进。这是敏捷开发的基本特征。敏捷开发强调结对编程，一个人写代码，一个人把握方向、思考下步工作。而且要求工作性质不同的人结对，如程序员与测试员，程序员与方案编写者，程序员与客售前或者售后，保证需求与代码一致。他也强调迭代，小范围的迭代，并且在迭代过程中实现代码重构。但是敏捷开发并不是一个简单的方法应用，他首先是一种思想，经过了长期研发工作，对研发质量，研发速度及研发效率有了深刻认识并深入思考，或许才能更好地理解。其次，他不是简单的开发思路，如要应用到企业研发，还需要精通敏捷开发的人指导或者是专业团队导入。应用敏捷开发本身就是一项大工程。第三，敏捷开发需要从上到下的支持，需要领导全方位支持，包括前几个月没有明显的效果，或许中间还会出现反复的情况。更需要团队成员的配合，按书上的话说是“团队认知高度统一”，就是了解并接受敏捷开发这种研发方式。如此说来，敏捷开发不可能随意应用到我们的研发中。不过，公司今年开展精益研发，说不定就用的是敏捷开发思路。因为敏捷开发本身就是精益研发方法。既然不能小范围地导入，那就学习一下他的一些思想吧。下面这些是在读的过程结合工作现状思考的几点。
产品经理：对于软件将要提供什么和为什么这是最重要的、项目团队值得花费时间的事情有着直观的理解。在微博上，有几个产品经理主要工作就是收集人们反馈的信息，当然这些产品经理都是搞网络产品的。我们的产品经理大多还站在项目经理的角度来管理产品，对产品的长期发展规划不足，对市场需求及反馈收集不足，却把主要精力用在方案与代码上面。利益攸关者：他们可能不参与项目的日复一日的开发，却对项目的成功扮演非常积极的角色，请他们参与每次的迭代演示，权衡需求方案等方面的重点最终使项目成功。我们的现有的作法就是评审，或许我们应该把评审参与人员范围扩大，如需求方面的评审应该叫上市场部或工程部。XP采用迭代而非阶段，所以在迭代过程中所有的人员都有参与、减少产品与目标的误差，从而实现目标。我们现在采用的就是阶段式研发，在产品发布后运用迭代来完善，但我们每次迭代周期过长，这是个很严重的问题。机会有的时候是会降临到等待的人头上，但那也只是争取它的人挑剩下的。在本书里能不时看到这种思辨性的句子。迭代，一般在一到三周。而我们的项目却都在一个月一上。长周期隐含了很多问题：迭代内容过多？研发质量太低？项目计划可控性差？这是今年工作的改进的重要入口。重构，改善代码质量的有效途径。书中没有对重构做详细的描述，网上是这样说的：就是在不改变软件现有功能的基础上，通过调整程序代码改善软件的质量、性能，使其程序的设计模式和架构更趋合理，提高软件的扩展性和维护性。是的，目标就是在提高软件的扩展性和维护性。这也是我们的弱项。也是下一步工作的重点。结对编程：找工作性质不同的两人结队，一个人写，一个人思考。我们可以一个人写方案和代码走读一个人实现代码和测试这种方法来实现结对编程。在研发结构和人员分工等各方面都合情合理。程序编写：初级程序员会创建费解的、拼凑的方案，他们本可以有机会学习更好的方法。这敏捷开发推崇结对编程的理由之一。我们的工作现在对此还不在意，也是我们产品问题多，代码不可重构的原因之一。降低软件开发成本的最好方法就是改善代码和设计的内在品质。部分分配对于生产力的不良影响极其可怕。这是本书在分析团队人员在团队中全职还是还是临时工时的一句话，可见临时工在国外也不受欢迎。如何做更少BUG？一，从测试驱动开发，把开发工作变成可验证的步骤；二，时间控制，结对编程（我们目前能做的是评审）；三，偿还技术债务（定期整理不稳定的模块）。我们不偿还技术债务直至问题出现。测试驱动开发（Test-Driven Development, TDD）由测试、编码、重构组成的快速循环，你的意图表达两次，通过不同的方式描述同一思想：第一次通过测试来描述，然后通过代码，如果它们相符，则很有可能它们的编码都正确，如果不相符，肯定是某处出现错误了。另：本书作写作者或许把本书作为一个项目来编写，每一小节都按固定的顺序，从问题、结论、禁忌、更多选择、延伸阅读这几个方面来写，不多不少，有理有据。
- 3、这是我第一本敏捷开发的书，内容全面，该介绍的都介绍了。产生了更多的疑问。1.如何理解XP中

《敏捷开发的艺术》

的各个角色，项目经理，产品经理等等2.从项目管理角度而言，各个角色的职责和范围是什么？3.结对编程是否真的适用？。。。【书摘】敏捷软件开发宣言：1.个人和交互胜过过程和工具2.可工作的软件胜过面面俱到的文档3.客户协作胜过合同谈判4.响应变化胜过遵循计划敏捷宣言背后的原则：1.我们的最高优先级是通过尽早的、持续的交付有价值的软件来满足客户。2.欢迎变化的需求，即使在开发的后期。敏捷过程利用变化为客户创造竞争优势。3.频繁交付能工作的软件，短则几周，长则几个月，时间间隔越短越好。4.项目开发以积极的个体为基础。为他们提供所需的环境和支持，并相信他们能完成工作。5.向开发团队传递信息或者只在开发团队内部传递信息，最高效、最有力的方法是面对面的交谈。6.能工作的软件是度量进度的首要标准。7.敏捷过程推动了可持续的软件开发。发起者、开发者和客户应该能长期维持一个恒定的速度。8.对技术卓越和良好设计的持续关注能增强敏捷能力9.简单性（将未完成的工作最大化的艺术）是非常重要的。10.最好的架构、需求和设计出自于自组织的团队。11.每隔一定时间，团队应该反思一下如何变得更有效，然后相应的调整或校正其行为。XP最令人瞠目结舌的一个主张就是你可以越过需求、设计、设计、测试阶段以及与之俱来的文档工作。但是，项目需要更多的需求、设计和测试--这就是为什么XP团队每天都从事这些活动。XP采用增量式设计和架构以持续地小步骤创建和改进设计。这一切都要归功于测试驱动开发（test-driven development, TDD）错综复杂地衔接起测试、编码、设计和架构。现场客户负责定义团队创建的软件。团队的其他成员能够并且应该贡献建议和主意，但是客户最终负责决定什么队与利益攸关者有价值。一般来说，产品经理、领域专家、交互设计师和需求分析师承担现场客户的角色。客户最重要的活动是发布计划（阐述项目愿景，发现特性和用户故事，管理风险等等）。也负责按照程序员的要求提供细节。

《敏捷开发的艺术》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com