

<<敏捷用户体验设计>>

图书基本信息

书名：<<敏捷用户体验设计>>

13位ISBN编号：9787111447214

10位ISBN编号：7111447212

出版时间：2014-1

作者：Diana DeMarco Brown

译者：姚军

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：<http://www.tushu007.com>

<<敏捷用户体验设计>>

内容概要

这是一本关于敏捷用户体验的专著，由资深用户体验设计专家撰写，不仅系统且深入地阐释将敏捷方法用于用户体验设计的工具、方法、原则和最佳实践，而且对各种常见的问题进行深入分析，包含大量实践案例，可操作性强，能为用户体验组织应用敏捷方法提供有效指导，让组织持续创造成功的产品和服务。

《敏捷用户体验设计：用户体验设计应用敏捷方法的技巧与最佳实践》共6章，第1章探讨《敏捷宣言》中传达的价值观和原则，这是敏捷方法的核心；第2章分析在团队完全认可敏捷的精神和价值观的前提下如何将用户体验融入敏捷环境；第3章通过丰富的案例详细介绍如何采用敏捷开发流程和设计思维，并从不同角度呈现参与者对敏捷用户体验的看法；第4章分析团队成功进行敏捷用户体验工作的因素，强调将重点放在团队对项目的注意力、健康的团队交流与沟通、确保足够的培训、持续重构和改进过程上才能打造高效的团队；第5章围绕在启动敏捷用户体验过程中遇到的“视情况而定”问题，如何将项目团队看作用户，确定他们的需求，创建一个优雅的解决方案；第6章分析用户体验团队如何在实践中吸收和采纳敏捷概念。

<<敏捷用户体验设计>>

书籍目录

《敏捷用户体验设计：用户体验设计应用敏捷方法的技巧与最佳实践》

译者序

前言

第1章 敏捷方法简介 1

1.1 引言 1

1.2 敏捷价值观+ux 3

1.2.1 个体和交互重于过程和工具 4

1.2.2 可用的软件重于完备的文档 5

1.2.3 客户协作重于合同谈判 6

1.2.4 对变化的响应重于遵循计划 7

1.3 敏捷原则+ux 8

1.3.1 原则1：我们最重要的任务是通过尽早和持续交付有价值的软件来满足客户 8

1.3.2 原则2：即使在开发的后期也欢迎需求的更改。

敏捷过程利用更改来为客户获得竞争优势 9

1.3.3 原则3：频繁交付可用软件，从几周到几个月，优先考虑较短的时间段 10

1.3.4 原则4：业务人员和开发人员在整个项目创作期间每天都必须一起工作 10

1.3.5 原则5：围绕积极的个体构建项目。

为他们提供环境并支持他们的需求，相信他们能够完成工作 11

1.3.6 原则6：在开发团队中传达信息的最有效方法是面对面的交谈 12

1.3.7 原则7：可用软件是进展的主要度量 13

1.3.8 原则8：敏捷过程提倡可持续开发。

项目方、开发人员和用户应该长期保持步调一致 14

1.3.9 原则9：对卓越技术和优秀设计的持续追求能够改进敏捷性 14

1.3.10 原则10：简洁性不可或缺，它是减少工作量的艺术 15

1.3.11 原则11：最好的架构、需求和设计源自于自我组织的团队 16

1.3.12 原则12：团队定期总结更为高效的手段，然后相应地调整自身的行为 17

1.4 常见方法 18

1.4.1 crystal 18

1.4.2 极限编程 18

1.4.3 scrum 19

1.4.4 混合敏捷 21

1.4.5 看板 22

1.4.6 scrumban 23

1.4.7 lean ux 24

1.5 常见术语 25

1.5.1 鸡和猪 25

1.5.2 产品负责人 25

1.5.3 scrum主管 26

1.5.4 冲刺 27

1.5.5 产品待办事项列表 27

1.5.6 用户故事 28

1.5.7 史诗 29

1.5.8 计划扑克 29

1.5.9 故事点估算 30

1.5.10 验收标准 31

<<敏捷用户体验设计>>

- 1.5.11 燃尽图 31
- 1.5.12 穿刺 32
- 1.5.13 agilefall 32
- 1.5.14 jeff patton 33
- 1.6 案例研究—jeff gothelf , theladders.com 33
- 关键点 36
- 1.7 小结 36
- 参考书目 36
- 第2章 敏捷方法+ux=敏捷ux 39
- 2.1 导言 39
- 2.2 将ux融入到敏捷环境中 41
- 2.3 ux工作 47
- 2.3.1 资源和人员 48
- 2.3.2 规格说明 51
- 2.3.3 用户调查 54
- 2.3.4 可用性测试报告 59
- 2.3.5 设计活动 62
- 2.4 案例研究—catherine robson , seachange international 64
- 关键点 68
- 2.5 小结 68
- 参考书目 69
- 第3章 案例研究 71
- 3.1 导言 71
- 3.2 suzanne o'neilly , appnexus 72
- 关键点 75
- 3.3 thyra rauch , ibm 76
- 关键点 78
- 3.4 archie miller , snagajob.com 78
- 关键点 82
- 3.5 carol smith , perficient 82
- 关键点 86
- 3.6 kayla block , par springer miller 86
- 关键点 90
- 3.7 无名氏1 , 一家企业级软件公司 91
- 关键点 93
- 3.8 christina york , ithaka 94
- 关键点 98
- 3.9 无名氏2 , 一家大型桌面软件公司 98
- 关键点 103
- 3.10 austin govella , avanande 103
- 关键点 107
- 3.11 josh o'connor , 爱尔兰国家盲人理事会 108
- 关键点 109
- 3.12 adrian howard , quietstars 109
- 关键点 111
- 3.13 elisa miller , ge healthcare的高级用户体验工程师 111
- 关键点 116

<<敏捷用户体验设计>>

- 3.14 小结 117
- 参考书目 117
- 第4章 共同的成功因素 119
 - 4.1 引言 119
 - 4.2 项目重于过程 121
 - 4.3 团队的活力 125
 - 4.4 沟通 127
 - 4.5 定义整体思路 130
 - 4.6 培训 131
 - 敏捷方法的学习资源 132
 - 4.7 适应和进化 136
 - 4.8 案例研究—sarah kahn , adzerk 137
 - 关键点 142
 - 4.9 案例研究—无名氏3, 一家专门从事产品直接营销的公司 142
 - 关键点 146
 - 4.10 小结 146
 - 参考书目 147
- 第5章 常见问题 149
 - 5.1 引言 149
 - 5.2 我们应该采用敏捷方法吗 149
 - 5.3 应该采用多长的冲刺周期 152
 - 5.4 ux应该制作什么可交付物 153
 - 5.5 ux团队如何融入开发团队的冲刺中 155
 - 5.6 如何在开发人员忙于实现设计时谈论另一个设计 156
 - 5.7 如果ux团队成员必须支持多于一个项目怎么办 157
 - 5.8 如何将用户调查融入冲刺周期 157
 - 5.9 如果团队声称是敏捷的,但是没有看到敏捷价值观的体现,该怎么办 158
 - 5.10 如果团队不在一起办公该怎么办 159
 - 5.11 当有人以“这不是敏捷方法”为由不做某些工作时,我该怎么办 160
 - 5.12 ux团队如何为下一次发行开展计划和调查 160
 - 5.13 如何管理内部利益相关方 161
 - 5.14 小结 162
 - 参考书目 162
- 第6章 将敏捷概念用于ux团队 163
 - 6.1 引言 163
 - 6.2 创建用户体验待办事项列表 163
 - 6.3 重复进行用户测试 164
 - 6.4 将工作分解为较小的部分 165
 - 6.5 不断的反馈和迭代 166
 - 6.6 反复进行的活动和仪式 166
 - 6.7 没有设计明星或者英雄 166
 - 6.8 与文档相比更注重沟通 168
 - 6.9 构思和传达用户故事 168
 - 6.10 定义验收标准 169
 - 6.11 减少预先设计 170
 - 6.12 小结 170
 - 折叠

<<敏捷用户体验设计>>

译者序

信息技术的巨变为我们呈现了一个丰富多彩、别开生面的世界。

软件行业的建立和发展归功于各相关门类技术和理论的发展，其中软件工程理论是核心和基础。

通过前人数十年的不懈努力，整个软开发生命期已经形成了一套理论体系，并诞生了许多不同的软件开发过程，其中被采用最多的是传统的瀑布开发模型和后起的敏捷开发模式。

瀑布开发模型起源于传统制造业，它提出了预先设计、精确定义需求、各阶段利用文档紧密衔接、反复迭代等思想，使软件制造真正成为一个严格的过程，并在许多大型项目上得到了普遍应用，在20世纪80年代成为软件开发的标准。

尽管软件业经历了许多变化，但是这一经典模型迄今仍经久不衰，在许多项目上发挥着巨大的作用。

随着网络的兴起和软件复杂度的提高，以及软件在人类生活等各个领域越来越明显的作用，以用户为中心的思想日益深入人心，人们的需求已经超出了传统的软件功能范畴，更多地强调使用的体验，交互设计、用户体验（UX）设计越来越成为蔚然成风的行当，这也给软件开发带来了深远的影响。

由于用户的反馈越来越成为软件开发和更新的依据，传统的开发模型推进缓慢、在各阶段可交付工件（例如文档）上开销过大、不能迅速适应变化的矛盾越来越突出。

在这种背景下，人们开始积极推行另一种更具备适应性、开发过程更迅速的开发模式——敏捷开发模式。

这种模式摒弃了传统模式下对预先确定需求和设计细节以及制作大量文档的要求，将软件分解为更小、更灵活的单元，可快速开发、测试、迭代，使得软件更容易根据用户的反馈做出改变，而对团队沟通的强调，更是这一方法的精髓，它不仅是一个开发过程，更是组织和团队文化的构建过程。

敏捷方法的价值观和准则不仅代表着新时代软件需求变化造成的影响，更代表着网络时代新经济实体高效协作的理念。

从用户体验设计和敏捷开发模式产生的背景来看，两者有许多融合之处，但是两者鲜明的特性，又使得真正的融合举步维艰。

由于敏捷开发模式是以开发活动为中心的，设计人员往往游离其外，而在传统过程中的大量习惯也影响了设计人员对敏捷开发模式的看法。

人们期待着敏捷UX方面的权威指导，期待着各个功能区真正地相互理解、协作和融合。

本书是用户体验设计专家Diana DeMarco Brown的力作，不仅为UX人员详细解说了敏捷开发过程的含义、核心价值观和准则，更宝贵的是提供了许多不同公司的案例，这些案例中有的团队已经在敏捷UX上达到了炉火纯青的地步，有些游离于传统方法和敏捷方法之间，还有些则面临着很大的挑战、苦苦挣扎，利用这些正面和反面的例子，读者可以看到不同情况下敏捷团队所需要面对和考虑的各种因素，从而深受启发。

在本书的翻译过程中，我们对敏捷开发模式和UX设计都有了新的认识，也期待着它能给更多的读者提供前进道路上的利器。

由于译者的水平有限，错误在所难免，敬请广大读者见谅。

本书的翻译工作主要由姚军完成，徐锋、陈绍继、郑端、吴兰陟、施游、林起浪、陈志勇、刘建林、宁懿等人也为本书的翻译工作做出了贡献，在此特别感谢机械工业出版社华章公司的编辑对翻译工作提出的中肯意见。

前言

当发现自己将要为一个使用Scrum的项目提供支持时，我立刻就变得小心翼翼。

我并非真的很了解Scrum和敏捷（Agile），但是我意识到，这些技术都在快速发展，而且没有真正地

<<敏捷用户体验设计>>

将UX考虑在内。

然而，我无法选择我的团队采用的开发过程，所以只能放下自己的不安，做好支持团队的准备。

我很幸运，有一些内部专家是用户体验（UX）团队的倡导者，他们和开发团队并行或者提前投入工作。

他们的模型看上去非常符合逻辑且顺畅，我很高兴应用这些技术。

但是从一开始，我就注意到自己所处的情况下有许多因素和他们的经验不相符，我不完全确定如何修改这些技术来适应环境。

我的团队在物理上不处于同一个位置（但是也不极端），规模大于推荐的标准，除了用户界面之外，该项目还有一个重要的基础结构组件。

这些问题都不是毁灭性的，但是我需要调整自己的过程，对于自己是否足够“敏捷”，我总是不太确定。

结束这个项目之后，我在本地的可用性专家会议上分享了自己的经验，这令人大开眼界。

人们渴望着这方面的深刻见解和指南，并且有着许多相同的问题。

然而，情况千变万化，这些问题的答案对每个人而言也各不相同。

我被不同的敏捷UX实现搞得筋疲力尽，觉得在围绕敏捷和UX的对话中，不足以反映出这种多样性。

我感觉到，如果我知道有许多方法可以实现敏捷UX，并且通过了解这些方法，我能够对自己的方法更加自信。

我想，“应该有人写本书”，几年之后，我做到了。

本书的目标是说明UX团队在敏捷上有许多成功和失败的方法，并阐明在不同的情况下，使用相同的策略可能造成不同的后果。

书中的案例研究说明有许多方法实现敏捷方法，UX团队在敏捷环境里需要不止一两年的时间才能做得好。

我研究了团队成功的原因，以及确定团队实现正面结果的最佳途径所需要考虑的因素。

在阅读本书之后，你将拥有确定特定情况下使用的敏捷UX方法所需的工具。

鸣谢

感谢DeMarco Interactive的Doug DeMarco为本书创作了出色的插图。

书摘

第1章

敏捷方法简介

1.1 引言

要理解敏捷形式的以用户为中心设计的方法，重要的是领悟敏捷的真实含义以及这一术语的出处。

由于敏捷技术有着深远而丰富的历史，并且持续地发展，它已经成为许多书籍、博客、白皮书、会议演讲和网站的主题，这些作品对于这一价值系统及其方法有各自的看法。

本章只涉及最常见的术语和概念，以及和用户体验方面最紧密相关的内容。

我鼓励大家花一些时间探索更多可用的资源，深入地理解这一思想以及在多年的发展中成长起来的应用该思想的不同方法。

理解实现敏捷设计没有唯一正确的途径也很重要。

从核心上，“敏捷”是一组在软件开发中对团队有价值的指南。

高效团队的总体目标是构建出色的软件，获得最终用户的赞誉。

与此相比，通过何种过程或者使用工具，以及这些过程和方法的应用方法都是次要的。

敏捷（agile）这个术语是从20世纪90年代寻找更好软件开发方法的过程中成长起来的。

<<敏捷用户体验设计>>

传统的方法（如瀑布方法）在当时已经令人觉得难以控制。

消费者期望软件有更高的质量和更多的功能，为了创建满足最终用户的产品，需要改变开发周期。

此外，开发周期需要适应和允许需求变化的现实。

从问题响应的角度来看，瀑布开发方法遇到的难题特别多，这主要是因为它天生无法在周期的早期阶段发现设计、架构或者代码本身的严重问题，直到发行周期的最后都不知道可能存在的问题，造成了低质量的产品或者更长的发行周期。

而且，传统方法更容易造成“竖井”，在这种环境下，产品经理将需求“从墙上”扔给设计人员，然后设计人员将其设计规范扔给开发团队，开发团队接着将他们的代码扔给最终批准产品发行的质量团队。

由于在可交付产品的创作期间，这些团队之间的交流沟通有限，每个团队都在玩电话游戏，并对规范和需求做出自己的解读。

正如电话游戏中那样，结果最终产品很少和最初的设计意图相符。

开发团队开始试验新技术，例如极限编程（Extreme Programming）、自适应编程（Adaptive Programming）、Scrum和其他方法，以便在不要求开发人员每天写24小时代码的情况下，找到创作高质量的软件并使其符合要求的更好途径。

2001年，这些思想的实施者在犹他州集会，创立了《敏捷宣言》（Agile Manifesto）。

这篇宣言及其附带的12个原则展示了这些方法试图实现的精神，对于考虑采用敏捷方法的机构来说是一个重要的开端。

理解“敏捷宣言”中表达的价值观是验证你自己的敏捷实现是否走上正轨的好办法。

虽然从2001年起这些方法、技术和术语已经有了发展，但是敏捷的核心价值观没有变化。

宣言中明确地表示：

“我们将通过实践和帮助其他人实践来发现更好的软件开发方法。

通过这项工作，我们已经达成如下价值观：

个体和交互重于过程和工具

可用的软件重于完备的文档

客户协作重于合同谈判

对变化的响应重于遵循计划

也就是说，后者仍有价值，但是我们应该更重视前者”

《敏捷软件开发宣言》，agilemanifesto.org

敏捷过程

1.2敏捷价值观+UX

《敏捷宣言》提供了最好的指南，是驱动敏捷组织的价值观的最简洁表达。

网站上提供了对《敏捷宣言》原则的描述，并提出了对宣言的更深刻理解，但是宣言本身已经很好地表达了核心思想。

引用这么久以前的作品似乎是种过时的做法，但是我发现，即使到现在，这些价值观仍和建立的时候同样切合实际。

从软件的角度看，2001年已经是很久以前了；即使不像中世纪那么久远，也相当于原子时代了。

毕竟，2001年是许多估值过高的网络公司破产、第一代iPod推出、微软发行Windows XP的一年，距离无所不在的Facebook推出也有好几年了。

今天的UX（用户体验）业者对那个年代完全没有记忆。

由于近几年敏捷方法处于舞台中心，这一主题产生了许多变种，因此假定它的发展已经超出原来的宣言也是合理的。

毕竟，有多少软件概念在10年之后还能行得通？

但是，这种假设是错误的。

原始敏捷运动的后续者仍然分享《敏捷宣言》中传达的核心价值观，只是采用不同的策略创建表达原始陈述精神的环境而已。

<<敏捷用户体验设计>>

在这些年之后，软件制造团队仍然挣扎于以里程碑交付产品为焦点的过程之中，而不是将眼光放在最终产品上。

《敏捷宣言》的意图是定义一个价值观系统，建立一个能够响应这些情况，同时发现团队成员个体价值，并创建高品质软件的文化。

《敏捷宣言》陈述了敏捷环境的核心价值，与此同时，许多技术能够构建一个过程和框架，支持这些价值观。

通过UX的放大镜来研究这些价值，能够说明两者之间存在着一种自然的关系。

遗憾的是，UX专家没有聚在一起，讨论出UX价值观的宣言和原则，并将其作为将大家团结在一起的标准。

实际上，尽管对以用户为中心设计的过程已经作了定义，但是除了“我们应该将用户需求作为设计过程的中心”这一思想外，并没有正式地表达其价值系统。

至于UX原则，就更加模糊了。

这并不是说没有任何UX原则，而是由许多人制定的UX原则各不相同。

人们自己杜撰UX设计原则，包括微软公司也定义了Windows的UX原则，并且在其网站上分享这些原则（参见<http://msdn.microsoft.com/enus/library/windows/desktop/dd834141.aspx>）。

UX原则的大部分定义倾向于提及保证用户参与过程这一使命，然后确定对用户有最大支持的其他指导方针。

我们将评估敏捷原则和价值观，并探索它们融入和支持UX业者参与的活动的活动的方法，而不是选择一种最喜爱的UX原则与敏捷原则进行比较，因为我发现任何一组原则都提供了值得思考的有趣内容，也还没有创建出另一组UX原则。

1.2.1 个体和交互重于过程和工具

尽管是《敏捷宣言》中表达的第一个价值观，但是它往往是团队在研究用于管理任务、用户故事（User Story）和生成燃尽图（Burn-Down Chart）的不同工具时最先被忘记的。

各种仪式引起的兴奋（如每日例会、代办事项梳理、计划扑克和回顾）以及合理召开这些会议的渴望，往往使团队忘记了这些仪式的真实意图——将重点放在人和沟通之上。

此外，探索和使用支持过程的新软件工具的乐趣，也很容易使团队的注意力转向错误的方向。

如果每日站立会议需要花费20分钟，而团队对简洁的状态报告习以为常，只要团队成员学习提供简洁状态报告的方法并且加强只分享相关信息的能力，这就没有问题。

尽管15分钟会议是一个很好的目标，但是它的要点并不是具体的时间，而是在足够短的会议中提供最少、最相关的信息，不给团队造成负担。

如果团队总是将这种会议持续30或者45分钟，那么这个问题就需要解决。

与此相似，如果计划扑克花费的时间太长、没有效率或者导致团队成员退出，那么就应该关注这不是团队所能使用的最佳技术，是否有更好的方法进行工作量估算。

虽然玩一组纸牌很有趣，但是这只是让大家参与工作量估算的一种方法，而且不是唯一的方法。

如果团队的周期中有多个冲刺（Sprint），而团队成员将更多的时间和精力花在任务管理工具上，团队就需要讨论不同的任务处理和跟踪方法。

如果在发行期间更换过程管理工具可能会引起混乱，进行一次关于工具的健康讨论能够找出问题和潜在的解决方案，帮助确定未来需要使用的不同工具。

敏捷方法中的一个重要部分是在整个过程中对其加以改善，而不是选择一种方法，然后不顾一切地坚持。

所有这些技术都是实现目的的手段；如果它们不能帮助团队增加和改进沟通，那么就应该重新看看《敏捷宣言》，思考怎样做才更为“敏捷”。

从用户体验的角度看，用与产品上相同的迭代设计技术对待这一过程的思路可能非常熟悉。

我们询问客户有关产品的反馈，并做出相应调整；产品团队是这一过程的消费者，应该可以对设计提

<<敏捷用户体验设计>>

出意见。

此外，设计人员和可用性专家习惯于考虑单独的最终用户而非一组功能；这本身有利于围绕用户故事的开发过程。

改进最终用户和系统之间的交互始终是我们最重要的关注点，毕竟，“交互设计人员”也是我们当中某些人职责的一部分。

采用敏捷方法，意味着将我们对待用户的眼光应用到队友和工作上。

有些UX人员很自然地将焦点放在关系之上。

由于UX团队通常不会自行在最终产品上实现这些设计，我们将依赖开发团队完成这一工作；因此，与开发团队和其他协作团队建立良好的关系始终是我们的最大兴趣。

然而，也可能发生团队和来自其他工作领域的同伴陷入“我们和他们”心态的情况。

实践敏捷方法是消除分裂心态，专注于团队建设的重大机遇。

好消息是，这一特殊的敏捷价值观允许和引导团队中的每个人将重点放在沟通和单独的团队成员身上，而不是专注于过程。

承认每个团队成员为团队带来独特的技能也是对传统过程的一个改变，在传统过程中，开发人员往往被看作可以互换的代码制造者。

珍视个人的技能，为承认用户体验团队成员带来的成果提供了机会，过程也可能为组成团队的个体作出相应的调整。

<<敏捷用户体验设计>>

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:<http://www.tushu007.com>