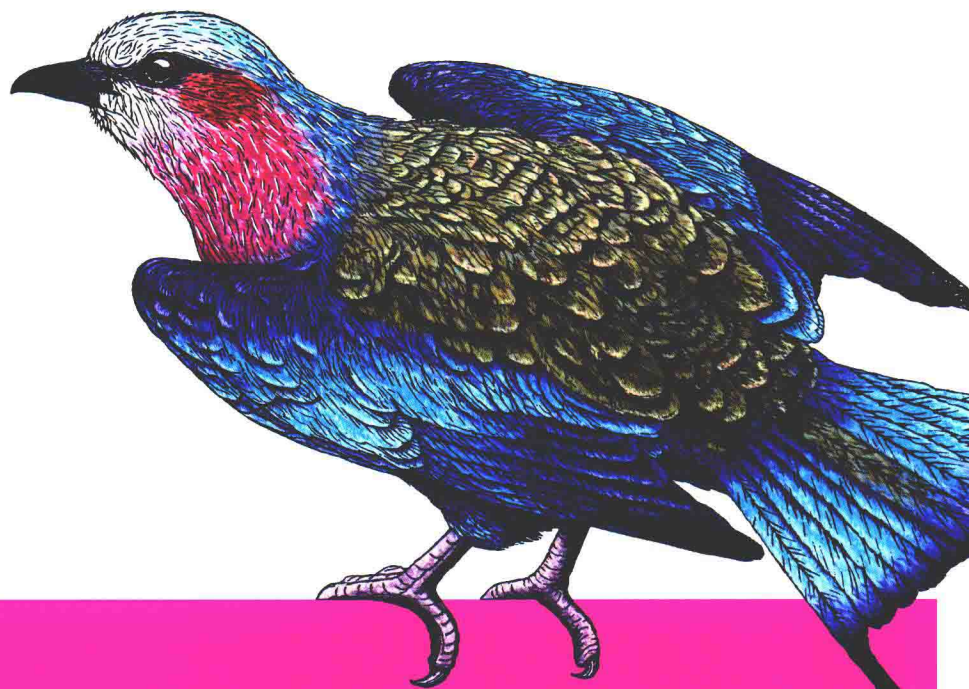


O'REILLY®



用户故事地图

User Story Mapping 增强团队协作, 洞察真实需求, 研磨优良产品

Jeff Patton 著
Martin Fowler Alan Cooper Marty Cagan 序
李涛 向振东 译

清华大学出版社

用户故事地图

Jeff Patton 著

李涛 向振东 译

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

O'REILLY®

O'Reilly Media, Inc. 授权清华大学出版社出版

清华大学出版社

北京

内容简介

用户故事地图作为一种有效的需求工具，越来越广泛地应用于开发实践中。本书以用户故事地图为主题，强调以合作沟通的方式来全面理解用户需求，涉及的主题包括怎么以故事地图的方式来讲用户需求，如何分解和优化需求，如果通过团队协作工作的方式来积极吸取经验教训，从中洞察用户的需求，开发真正有价值的、小而美的产品和服务。

本书适合产品经理、用户体验设计师、产品负责人、业务分析师、IT项目经理、敏捷教练和精益教练阅读和参考，也更适合作为企业培训手册，打造高效能的团队协作能力。

Copyright © 2014 Jeff Patton. All rights reserved.

Authorized Simplified Chinese translation edition, by O'Reilly Media, Inc., is published by Tsinghua University Press, 2016. Authorized translation of the original English edition, 2014 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

本书之英文原版由O'Reilly Media, Inc.于2014出版。

本中文简体翻译版由O'Reilly Media, Inc.授权清华大学出版社于2016年出版。此翻译版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc.的许可。

版权所有，未经书面许可，本书的任何部分和全部不得以任何形式复制。

北京市版权局著作权合同登记 图字：01-2015-2861

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目(CIP)数据

用户故事地图. (美) 帕顿 (Patton, J) 著; 李涛, 向振东译. --北京: 清华大学出版社, 2016

书名原文: User Story Mapping

ISBN 978-7-302-42994-4

I. ①用… II. ①帕…②李…③向… III. ①产品开发 IV. ①F237.2

中国版本图书馆CIP数据核字(2016)第031106号

责任编辑：文开琪

封面设计：Ellie Volckhausen, 张健

责任校对：周剑云

责任印制：沈露

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦A座 邮政编码：100084

社总机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62795954, jsjic@tup.tsinghua.edu.cn

质量反馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者：北京亿浓世纪彩色印刷有限公司

经 销：全国新华书店

开 本：178mm×233mm 印 张：16.25 字 数：238千字

版 次：2016年4月第1版 印 次：2016年4月第1次印刷

印 数：1~3000

定 价：59.00元

目录

Martin Fowler序	1
Alan Cooper序	3
Marty Cagan序	5
前言	9
致谢	17
使用前必读.....	21
第1章 产品全景图	35
让我们从头开始	35
故事是讲出来的，不是写出来的	36
讲故事，要完整	37
Gary的悲剧.....	38
边讲边记	39
创意框架	40
刻画用户画像	41
讲用户的故事	42
探索细节和可选项.....	45

第2章 计划，为了更少的开发	51
故事地图帮助大型组织建立共识	52
创建故事地图的过程可以帮助发现设计中的坑	54
要做的总是太多	55
划分MVP发布计划	56
划分发布路线图	57
为成果排列优先级，而非功能	57
这是魔法吗？没错	58
为什么要反复讨论MVP	60
MVP根本就不是产品	61
第3章 计划，为了更快的学习	63
从讨论机会开始	64
验证问题	64
在设计原型过程中学习	65
要能够质疑用户所说的内容	66
在开发过程中学习	66
迭代直至可行	68
错误的做事方式	69
基于验证的学习	70
真正的最小化试验	71
重点复述	71
第4章 计划，为了按时发布	73
要让团队所有成员都清楚	74
估算的秘密	75
制定可逐步达成的开发计划	76
不要将所有的迭代产出都对外发布	77
关于估算的另外一些秘密	77
管理研发预算	78
迭代与增量	82
开局、中局和末局策略	82

根据开发策略切分故事地图.....	83
都是关于风险.....	84
“剧透”第5章主题.....	84
第5章 如何创建故事地图.....	85
1. 分步骤写出你的故事.....	85
2. 组织情节.....	88
3. 探索替代故事.....	89
4. 提取故事地图的主干.....	91
5. 切分出能帮你达成特定目标的任务.....	92
就是这样简单！你已经学会了所有重要概念.....	93
请在家里或者办公室里练习.....	94
这张地图是现在的，不是将来的.....	95
实操案例.....	96
练习容易，落地难.....	97
故事地图仅仅只是个开始.....	98
第6章 用户故事的故事.....	103
Kent Beck的创意.....	103
简单的事情并不一定容易做到.....	104
Ron Jeffries的3C原则.....	105
文字和照片.....	107
小结.....	108
第7章 如何把故事讲得更好.....	109
Connextra公司的用户故事模板.....	109
模板僵尸和万能犁.....	113
提升讨论效果的检查单.....	115
创建度假照片.....	117
需要操心的事情还多着呢.....	117
第8章 不要把所有内容都写在卡片上.....	119
不同角色，各有所需.....	119

我们需要一张更大的故事卡	120
信息辐射器和信息冰箱	122
错误的工具和错误使用工具	124
第9章 卡片只是个开始	129
在头脑中构建清晰的图像	130
养成口述用户故事的习惯	130
检视产出	131
你又不是用户	132
开发过程就是学习的过程	133
不仅仅是软件	134
为学习做计划，学习如何做计划	134
第10章 做产品好比烤蛋糕	135
食谱	135
切分大蛋糕	137
第11章 碎石行动	141
故事的大小很重要	141
把故事比喻为石头	142
史诗故事是大石头，有时可以用来攻击他人	144
用主题来组织故事	145
忘掉这些术语，专注于讲故事	145
从机会开始	146
探索最小可行方案	147
在交付阶段深入每个故事的细节	148
在开发过程中保持日常对话	150
评估每一份产出	151
与用户和客户一起评估	152
与业务干系人一起评估	152
发布和持续评估	153

第12章 谁是碎石负责人	155
有价值的-可用的-可行的.....	156
一个成功的探索团队需要更多的人参与	158
神勇三蛟龙.....	159
产品负责人好比音乐制作人.....	162
这项工作并不简单.....	163
第13章 从机会开始	165
针对机会展开对话.....	165
深入挖掘机会，丢弃机会或思考机会.....	166
机会不应该是一种委婉的说法	170
故事地图和机会	170
挑剔.....	176
第14章 通过探索来建立共识	177
探索不是开发软件.....	177
探索的4个核心步骤.....	178
探索活动、讨论和工件	191
探索的目的是建立共识	192
第15章 通过探索来进行验证性学习	195
大多数时候，我们其实都是错的	195
糟糕的往事.....	196
同理，聚焦，形成想法，制作原型，测试	197
如何把好事弄糟	200
短期验证学习循环.....	201
精益创业思想改变产品设计.....	202
故事和故事地图呢.....	206
第16章 提炼、定义和开发.....	209
卡片，对话，更多卡片，更多对话.....	209
细分和提炼.....	209
故事工作坊.....	210

在冲刺或迭代计划阶段开展故事对话	213
人人参与并非明智之举	215
分解和瘦身	217
如何在交付阶段使用故事地图	222
如何使用故事地图来可视化进展	222
在故事工作坊中使用简易地图	223
第17章 故事呢，就好比《行星战机》	229
把碎石子儿重新聚集起来	230
地图绘制要适度	232
千万不要小题大作	233
第18章 开发完成后怎么学习	235
团队回顾	235
和团队外的角色一起回顾	238
够用	240
向用户学习	241
从发布中学习	242
预定计划中的结果	242
使用故事地图来评估发布是否准备就绪	243
结语	245

Martin Fowler序

敏捷软件开发运动的兴起为软件行业带来了诸多积极的变化，大型需求要进行拆分，这个意识的建立便是其中之一。切分后的需求称为“故事”（story），故事的使用使得软件开发项目的过程进一步可视化。通过故事方式来组织开发的产品，每一个故事实现都和整个软件完全集成，每个人都可以看到产品在不断成长。用户也可以理解故事，开发者通过决定下一个迭代开发哪个故事来管理软件开发项目。可视化程度的大幅提升，使得用户可以深入参与到项目中来，而不是像以前那样需要等上一年甚至更久时间才能拿到开发完成的新特性。

需求拆分本身也有很多负面影响，容易丢失软件系统全景图（whole picture）便是其中之一。开发工作进展到后期，你可能得到的是一堆无法拼接在一起的碎片。也可能由于过度陷入细节而丢掉用户诉求的本质，最终构建出用户不需要的产品。

故事地图是一门在需求拆分过程中保持全景图的技术。

如果要用一句话来诠释本书的话，非上面这句话莫属了，这句话本身就很有价值。全景图可以帮助团队和用户有效的沟通，帮助参与其中的人避免开发非必要的特性，也为一致的用户体验提供了基准。当我询问ThoughtWorks（思特沃克）的同事如何开发用户故事时，他们最常提到的核心技术就是用户故事地图。这些ThoughtWorks同事也是在Jeff（杰夫）的工作坊学到这门技术的，因为Jeff开发了故事地图，也只有他能将故事地图讲到如此淋漓尽致的程度。Jeff写这本书正是为了帮助读者直接从源头学习这门技术。

但是，这本书并非单是为那些名片印着产品经理、业务分析师头衔或者在线简历中写着产品经理头衔的人而写的。在采用功能敏捷开发方法的十年中，最让我失望的一点是，程序员把故事当作和产品经理之间进行沟通的单行道。在最初的时候，故事的目的是

激发沟通中的火花。要想开发出能有效支撑用户活动的软件，就要求助于开发软件中最关键的角色，因为只有程序员最清楚软件可以做什么。程序员需要理解用户想要达成的目标，需要在前期捕获用户需求的阶段就参与进来，一起开发故事。懂得故事地图技术的程序员能更好地理解用户上下文，在软件成形期间更好地参与进来，从而取得更好的工作成果。

Kent Beck（肯特·贝克，最早提出用户故事概念的人）发展了自己在软件开发方面的理念，他呼吁团队把沟通作为高效团队的核心价值。故事，是程序员和其他角色沟通中的必备要素，故事地图对这些要素组织为结构化，以此来强化软件开发中最关键的部分——沟通。

——Martin Fowler（马丁·福勒）

2014年6月18日

Alan Cooper序

在Mary Shelley（玛丽·雪莱）的著名科幻小说《科学怪人》中，疯狂的弗兰肯斯坦博士用尸体碎片创造了一个生命，那时候电力还被视作一项新技术，弗兰肯斯坦博士用电力给生物注入生命。当然，这只是小说中的情节，在现实世界中使用尸体碎片创造生命实际上根本就不可能。

然而在软件开发中，我们一直在试图这样做。在软件中堆砌一个又一个新功能，然后陷入“为什么没有多少用户喜欢这个产品”的困惑中。这个谜题的核心在于开发人员将工程方法作为了设计工具，实际上两者不是一回事儿。

程序员逐个开发特性是完全合情合理的，并且经过数年验证是一个行之有效的策略。同样经过数年验证的是，设计软件产品的行为和范围时，也遵循逐个进行的方式，这就有点像科学怪人的做事方式了。

尽管有相通之处，设计软件行为和开发软件的实践之间其实有明显的不同，主要原因在于这两件事是由不同技能特长的人来承担的。像交互设计师那样花好几个小时的时间观察用户行为和提取行为模式，这样的工作会让程序员抓狂。而像程序员那样花几个小时研究算法，对大多数设计师而言也同样难以忍受。

但是，当设计和开发两类工作产生协同的时候，就会像弗兰肯斯坦所使用的电力一样，能够创造出有生命的产品。团队协作为产品注入生命力，并让用户也爱上它。

虽然协作本身并不是什么新概念，但要做到高效协作实际上确实十分困难。程序员工作的节奏、语言和交互设计师之间有非常大的差别。

程序员和设计师在各自的领域中都非常专业、精干，都有自己的工作规范，同时他们又

有着共同的弱点。设计问题是很难用开发术语来描述的，同样，开发难题也难以使用设计术语来说明白。这对姊妹学科之间缺乏共同的语言，而连接两者恰恰是Jeff Patton（杰夫·帕顿）所擅长的。

Jeff的用户故事地图方法能够为程序员所理解，同样也可以为设计师所理解。用户故事地图就像数字时代的罗塞塔石碑（Rosetta Stone）。

撇开业界对敏捷的成见，敏捷软件开发方法本身也不见得是一种良好的产品设计方法。敏捷开发是一种很好的思维方式，可以使设计方案更顺利地交付，却无法产出能让用户喜爱的产品。换句话说，我们看到过不少优秀的设计，文档完成后交给开发人员，不管是敏捷开发还是非敏捷开发，设计的核心理念在实现过程中都会被抹杀掉。

Jeff Patton的用户故事地图方法是连接开发和设计的桥梁。交互设计的核心是发现用户行为并像讲故事一样把它们描述出来。软件开发则是将这些描述拆分、实现并集成到产品中。在这个复杂的过程中，设计的核心理念非常容易丢失。是的，就像是手术失败，所有的规定操作都做完，病人最后却死在手术台上。

通过用户故事地图的方式来处理用户故事，设计仍然保持其叙事结构，开发工作也可以得到很好的分解从而得以高效实现。设计师的方案以规范化的用户故事形式描述，在开发过程中流动并保持其完整性。

在传统公司中已经证实，以两三百人规模的团队，要开发出能让用户喜爱的产品几乎是不可能的。而创业社区则证实，四五个人组成的创业团队可以开发出能让人们喜爱的小产品，即使这些小产品也会最终随着规模变大而失去光芒。我们面临的挑战是如何创造出用户喜爱的大型软件产品。大型软件产品用户群广，并且用户从事的是复杂的商业活动。想把这样的软件做得有趣并且简单易学，是非常困难的。

要想避免将大型软件产品开发成“科学怪人”，唯一的方法是学习如何充分协调好产品设计和软件开发。在这方面，没有人比Jeff Patton做得更好。

——Alan Cooper（艾伦·库珀）

2014年6月17日

Marty Cagan序

我的职业生涯非常幸运，因为我一直有机会和世界顶尖的许多产品技术团队合作。他们创造出用户非常喜爱，并且每天都在使用的产品。这些团队正在改变世界。

我也曾经受命前往帮助一些做得不那么好的公司。创业团队努力在钱烧完之前找到新的投资。大公司则挣扎于复制早期的成功。团队无法持续为业务贡献价值。主管则为新想法何时才能上线操碎了心。工程师对产品经理满腹怨言。

在这个过程中，我认识到顶尖产品公司在软件设计开发上和普通公司之间存在巨大的差别。从主管行为到团队授权级别；到团队协作的方式；到组织在投资、招聘和产品开发方面的思考；到文化；再到产品、设计、开发如何协作共同发现对客户行之有效的解决方案。

这本书的题目是用户故事地图，但仔细阅读之后，你会发现这本书的内容并不局限于故事地图这一强有力却看似很简单的技术上。这本书更多讲述团队如何沟通、协作并最终交付好的产品。

大部分人是没有机会近距离观察一个强大的产品团队是如何运作的。读者能够了解的更多是自己公司以及前东家是如何运作的。所以接下来我会帮助大家来识别顶尖产品团队和普通团队之间的差别。

我非常赞同Ben Horowitz（本·霍罗威茨）的文章“好的产品经理和差的产品经理”中的观点，下面借用其形式，初步探讨一下好的产品团队和差的产品团队之间的主要不同。

好的团队，有引人入胜的产品愿景，怀着传教士般的热忱在工作。差的团队，像是由雇佣兵组成的，当一天和尚撞一天钟，靠混的。

好的团队，从关键业务指标得到启发，通过观察用户的痛点和分析用户使用过程中产生的数据，不断尝试新技术解决现实问题。差的团队，从销售人员和用户那里收集需求。

好的团队，理解谁是主要干系人，干系人所受的约束，承诺引入解决方案，方案对用户和客户有用，同时也满足业务上的约束条件。差的团队，只知道从干系人那里收集需求。

好的团队，掌握大量的技术手段，这些技术手段可以快速验证哪些产品创意是值得开发的。差的团队，召集会议来制定路标和排列优先级。

好的团队，喜欢和公司内有想法的主管展开头脑风暴和讨论产品。差的团队，在团队之外的人胆敢提议他们做任何事的时候，会觉得自己受到了冒犯。

好的团队，产品经理、交互设计师和开发工程师坐在一起，对功能、用户体验、技术可行性达成一致见解。差的团队，各自坐在小格子间里，没有文档和会议安排，就不会主动响应其他人的请求。

好的团队，持续尝试新想法以求创新，过程中会注意保护公司利益和品牌。差的团队，仍然坐着等待开始尝试的指令。

好的团队，对于创造出成功产品所需的技能很有信心，比如强大的交互设计能力。差的团队，甚至压根儿就不知道交互设计为何物。

好的团队，保证开发工程师每天有时间参与产品原型的讨论，为做好产品献计献策。差的团队，在迭代计划会上展示原型，一心只为了估出工作量。

好的团队，每周直接和用户交流，以更好地理解用户诉求，并试探用户对最新的产品创意的反馈。差的团队，以为他们自己就能代表用户。

好的团队，清楚地知道尽管他们很喜欢自己在产品上的创意，但这些创意中的很大一部分用户并不见得会接受，即使有一些被用户接受了，也需要经过多个迭代的打磨才能达到预期的效果。差的团队，只开发路标上规划的内容，能按时交付，只求不出重大质量问题就阿弥陀佛。

好的团队，理解速度和快速迭代对于产品创新的价值，更知道速度来自于正确的方法，而非强制加班。差的团队，抱怨同事工作不够努力，速度太慢。

好的团队，在评估方案，确认可行并对用户和业务有实际价值后，共同做出承诺。差的团队，抱怨自己公司是一个受销售驱动的公司。

好的团队，使用工具，以便快速了解用户是如何使用产品的，并基于数据做出判断。差的团队，认为统计分析是可有可无的。

好的团队，持续集成和发布产品，因为他们知道持续的小发布能为用户提供了稳定可靠的解决方案。差的团队，在经过痛苦的集成联调之后，手工测试，一次性发布所有功能。

好的团队，专注于用户。差的团队，专注于竞品。

好的团队，在关键业务目标重大影响达成后庆祝。差的团队，在终于发布产品之后庆祝。

我已经意识到读者会困惑，上面所提到的这些东西和用户故事地图有什么关系呢？我知道你会感到惊讶。这恰恰就是我成为故事地图铁杆儿粉丝的原因。

在我接触过的敏捷专家中，真正有能力帮助产品团队提升产品研发能力水平的并不多，Jeff Patton便是其中之一。我观察了Jeff在产品发现（Product Discovery）阶段亲自动手和团队一起工作的场景。我也把他介绍给公司，因为他做事非常高效。团队也很喜欢他，因为他不但知识丰富，人也非常幽默。

产品经理都聚到一起写需求文档，交互设计师忙于为产品进行涂脂抹粉般的设计，工程师躲在地下室写代码，这样的日子在顶尖团队中早已经销声匿迹。现在，是时候把这些现象清出你的团队了。

——Marty Cagan（马蒂·卡根）

2014年6月18日

