

学习敏捷 - 构建高效团队

作者：[美] Andrew Stellman

版权信息

书名：学习敏捷：构建高效团队

作者：[美] Andrew Stellman, Jennifer Greene

译者：段志岩 郑思遥

ISBN：978-7-115-44755-5

本书由北京图灵文化发展有限公司发行数字版。版权所有，侵权必究。

您购买的图灵电子书仅供您个人使用，未经授权，不得以任何方式复制和传播本书内容。

我们愿意相信读者具有这样的良知和觉悟，与我们共同保护知识产权。

如果购买者有侵权行为，我们可能对该用户实施包括但不限于关闭该帐号等维权措施，并可能追究法律责任。

091507240605ToBeReplacedWithUserId

[版权声明](#)

[O'Reilly Media, Inc. 介绍](#)

[业界评论](#)

[本书赞誉](#)

[序](#)

[前言](#)

[致谢](#)

[Safari® Books Online](#)

[联系我们](#)

[电子书](#)

[第 1 章 学习敏捷](#)

[1.1 什么是敏捷](#)

[1.2 本书的读者对象](#)

[1.3 本书的目标](#)

[1.4 努力建立敏捷思维](#)

[1.5 本书结构](#)

[第 2 章 理解敏捷价值观](#)

[2.1 团队主管、架构师和项目经理走进了一间酒吧.....](#)

[2.2 没有银弹](#)

[2.3 敏捷可以拯救乱局吗](#)

[2.3.1 引入敏捷，带来变化](#)

[2.3.2 “聊胜于无”的结果](#)

[2.4 视角割裂](#)

[2.4.1 视角割裂带来的问题](#)

[2.4.2 为什么视角割裂只能做到“聊胜于无”](#)

[2.5 敏捷宣言帮助团队认识实践的目的](#)

[2.5.1 个体和互动高于流程和工具](#)

[2.5.2 可工作的软件高于详尽的文档](#)

[2.5.3 客户协作高于合同谈判](#)

[2.5.4 响应变化高于遵循计划](#)

[2.5.5 原则高于实践](#)

[2.6 理解敏捷的“大象”](#)

[让实践快速上手](#)

[2.7 着手采用一套新方法](#)

[第3章 敏捷原则](#)

[3.1 敏捷软件开发的12条原则](#)

[3.2 客户总是对的吗](#)

[“按我现在说的做，而不是按我之前说的做”](#)

[3.3 交付项目](#)

[3.3.1 原则1：最优先要做的是尽早、持续地交付有价值的软件，让客户满意](#)

[3.3.2 原则2：欣然面对需求变化，即使是在开发后期。敏捷过程利用变化为客户维持竞争优势](#)

[3.3.3 原则3：频繁交付可工作的软件，从数周到数月，交付周期越短越好](#)

[3.3.4 改进电子书阅读器团队的项目交付计划](#)

[3.4 沟通和合作](#)

[3.4.1 原则4：在团队内外，面对面交谈是最有效、也是最高效的沟通方式](#)

[3.4.2 原则5：在整个项目过程中，业务人员和开发人员必须每天都在一起工作](#)

[3.4.3 原则6：以受激励的个体为核心构建项目，为他们提供环境和支持，相信他们可以把工作做好](#)

[3.4.4 在电子书阅读器项目中采用更好的沟通方式](#)

[3.5 项目实施——推进项目](#)

[3.5.1 原则7：可工作的软件是衡量进度的首要标准](#)

[3.5.2 原则8：敏捷过程倡导可持续开发。赞助商、开发人员和用户要能够共同、长期维持其步调，稳定向前](#)

[3.5.3 原则9：坚持不懈地追求技术卓越和设计优越，以此增强敏捷的能力](#)

[3.5.4 改善电子书阅读器团队的工作环境](#)

[3.6 项目和团队的持续改进](#)

[3.6.1 原则10：简单是尽最大可能减少不必要工作的艺术，是敏捷的根本](#)

[3.6.2 原则11：最好的架构、需求和设计来自自组织的团队](#)

[3.6.3 原则12：团队定期反思如何提升效率，并依此调整](#)

[3.7 敏捷项目：整合所有原则](#)

[第4章 Scrum和自组织团队](#)

[4.1 Scrum的规则](#)

[4.2 第1幕：Scrum的适用条件](#)

[4.3 Scrum团队中每个人都要对项目负责](#)

[4.3.1 Scrum主管指导团队的决策](#)

[4.3.2 产品所有者帮助团队了解软件的价值](#)

[4.3.3 每个人都对项目负责](#)

[4.3.4 Scrum有一组自己的价值观](#)

[4.4 第2幕：状态更新只是社交网络的玩法8](#)

[4.5 整个团队参与每日Scrum例会](#)

[4.5.1 反馈和“可见 - 检查 - 调整”周期](#)

[4.5.2 最后责任时刻](#)

[4.5.3 召开有效的每日Scrum例会](#)

[4.6 第3幕：将冲刺计划写到墙上](#)

[4.7 冲刺、计划和回顾会议](#)

[4.7.1 迭代式与增量式](#)

[4.7.2 冲刺成也在于产品所有者，败也在于产品所有者](#)

[4.7.3 可见性和价值观](#)

[4.7.4 计划并执行有效的Scrum冲刺](#)

[4.8 第4幕：尽力之后](#)

[第5章 Scrum计划和集体承诺](#)

[5.1 第5幕：出乎意料](#)

[5.2 用户故事、速度和普遍接受的Scrum实践](#)

[5.2.1 提升软件价值](#)

[5.2.2 以用户故事构建用户真正会用到的功能](#)

[5.2.3 满意条件](#)

[5.2.4 故事点和速度](#)

[5.2.5 燃尽图](#)

[5.2.6 通过用户故事、故事点、任务和任务板来计划并实施冲刺](#)

[5.2.7 广受认可的Scrum实践](#)

[5.3 第6幕：第一次胜利](#)

[5.4 回顾Scrum价值观](#)

[5.4.1 具体实践没有价值观也有效果（只是别管它叫Scrum）](#)

[5.4.2 你的公司文化与Scrum的价值观兼容吗](#)

[第6章 极限编程与拥抱变化](#)

[6.1 第1幕：开始加班](#)

[6.2 极限编程的主要实践](#)

[6.2.1 编程实践](#)

[6.2.2 集成实践](#)

[6.2.3 计划实践](#)

[6.2.4 团队实践](#)

[6.2.5 为什么开发团队抵制变化，上述实践如何提供帮助](#)

[6.3 第2幕：计划有变，但我们还是看不到希望](#)

[6.4 极限编程的价值观帮助团队改变心态](#)

[6.4.1 极限编程帮助开发人员学会与用户协作](#)

[6.4.2 开发团队的怀疑会破坏实践的效用](#)

[6.5 正确的思维从极限编程的价值观开始](#)

[6.5.1 极限编程的价值观](#)

[6.5.2 以善意铺就](#)

[6.6 第3幕：势头的变换](#)

[6.7 理解极限编程价值观，拥抱变化](#)

[6.7.1 极限编程的指导原则](#)

[6.7.2 极限编程指导原则可以加深对计划的理解](#)

[6.7.3 极限编程指导原则与实践相互促进](#)

[6.7.4 反馈循环](#)

[第7章 极限编程、简化和增量式设计](#)

[7.1 第4幕：再次加班](#)

[7.2 代码和设计](#)

[7.2.1 代码异味和反模式（如何判断你是不是聪明过头了）](#)

[7.2.2 极限编程团队主动寻找和修复代码异味](#)

[7.2.3 钩子、边界情况以及功能过多的代码](#)

[7.2.4 代码异味会增加复杂性](#)

[7.3 把编码和设计决定留到最后责任时刻](#)

[7.3.1 决然重构，偿还技术债务](#)

[7.3.2 持续集成，排查设计问题](#)

[7.3.3 避免一体式设计](#)

[7.4 增量式设计 with 极限编程的整体实践](#)

[7.4.1 有时间进行思考，团队才能做好工作](#)

[7.4.2 团队成员彼此信任并共同作出决定](#)

[7.4.3 极限编程的设计、计划、团队和整体实践形成了一个带动创新的系统](#)

[7.4.4 增量式设计与为了复用而设计](#)

[7.4.5 简化单元交互，系统实现增量式成长](#)

[7.4.6 优秀的设计源自简单的交互](#)

[7.5 第5幕：最终得分](#)

[第8章 精益、消除浪费和着眼全局](#)

[8.1 精益思维](#)

[8.1.1 你已经理解了很多精益价值观](#)

[8.1.2 承诺、选择意识和集合式开发](#)

[8.2 第1幕：还有一件事.....](#)

[8.3 创造英雄与神奇思维](#)

[8.4 消除浪费](#)

[用价值流示意图发现浪费](#)

[8.5 加深对产品的理解](#)

[8.5.1 着眼全局](#)

[8.5.2 找到问题的根本原因](#)

[8.6 尽快交付](#)

[8.6.1 使用面积图可视化工作进度](#)

[8.6.2 限制进行中的工作，控制瓶颈](#)

[8.6.3 拉动式系统帮助团队消除约束](#)

[第9章 看板方法、流程和持续改进](#)

[9.1 第2幕：紧赶慢赶的游戏](#)

[9.2 看板方法的原则](#)

[9.2.1 找到一个出发点并由此进行实验性的演进](#)

[9.2.2 用户故事进去，代码出来](#)

[9.3 用看板方法改进流程](#)

[9.3.1 将工作流程可视化](#)

[9.3.2 限制进行中的工作](#)

[9.4 测量并管理流量](#)

[9.4.1 用CFD和进行中工作面积图测量并管理流量](#)

[9.4.2 用利特尔法则控制系统的流量](#)

[9.4.3 用进行中工作上限管理流量，自然地创造缓冲](#)

[9.4.4 让过程策略明确统一](#)

[9.5 看板方法下自然发生的行为](#)

[第10章 敏捷教练](#)

[10.1 第3幕：还有一件事（又来了?!）.....](#)

[10.2 教练要理解人们为什么不想改变](#)

[教练会留意团队抗拒变化的征兆](#)

[10.3 教练要理解人们如何学习](#)

[使用“守破离”帮助团队学习某种方法的价值观](#)

[10.4 教练清楚如何让一套方法起作用](#)

[10.5 进行敏捷指导时的原则](#)

[关于作者](#)

[关于封面](#)

版权声明

©2015 by Andrew Stellman, Jennifer Greene.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and Post & Telecom Press, 2017. Authorized translation of the English edition, 2015 by 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.

英文原版由 O'Reilly Media, Inc. 出版，2015。

简体中文版由人民邮电出版社出版，2017。英文原版翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权所有者的许可。

版权所有，未得书面许可，本书任何部分和全部不得以任何形式重制。

O'Reilly Media, Inc. 介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 *Make* 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过图书出版、在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar 博客有口皆碑。”

——*Wired*

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——*Business 2.0*

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——*CRN*

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——*Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路遇到岔路口，走小路（岔路）。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——*Linux Journal*

欢迎访问：电子书学习和下载网站（<https://www.shgis.cn>）

文档名称：《学习敏捷 - 构建高效团队》[美] Andrew Stellman 著.epub

请登录 <https://shgis.cn/post/439.html> 下载完整文档。

手机端请扫码查看：

