

O'REILLY®

TURING

Hello, Startup

# 奔跑吧, 程序员



从零开始打造产品、技术和团队

[美] 叶夫根尼·布里克曼 ○著 吴晓嘉○译



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

**TURING**

# 奔跑吧，程序员： 从零开始打造产品、技术和团队

---

## Hello, Startup

[美] 叶夫根尼·布里克曼 著  
吴晓嘉 译

Beijing • Boston • Farnham • Sebastopol • Tokyo

**O'REILLY®**

O'Reilly Media, Inc. 授权人民邮电出版社出版

人民邮电出版社  
北 京

## 图书在版编目 (CIP) 数据

奔跑吧,程序员:从零开始打造产品、技术和团队 /  
(美)叶夫根尼·布里克曼 (Yevgeniy Brikman) 著;吴  
晓嘉译. — 北京:人民邮电出版社, 2018.7  
ISBN 978-7-115-48366-9

I. ①奔… II. ①叶… ②吴… III. ①程序设计  
IV. ①TP311.1

中国版本图书馆CIP数据核字(2018)第087212号

## 内 容 提 要

本书以软件工程师出身的创业者角度,全面介绍了创业公司该如何打造产品、实现技术和建立团队,既是为创业者打造的一份实用入门指南,又适合所有程序员系统认识IT行业。书中内容分为三部分——技术、产品和团队,详细描绘创业的原始景象,具体内容包括:创业点子、产品设计、数据与营销、技术栈的选择、整洁的代码、软件交付、创业文化、招兵买马,等等。

本书适合所有程序员,尤其是准备创业的技术人员。

- 
- ◆ 著 [美]叶夫根尼·布里克曼  
译 吴晓嘉  
责任编辑 朱巍  
责任印制 周昇亮
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
三河市君旺印务有限公司印刷
  - ◆ 开本: 700×1000 1/16  
印张: 23.5  
字数: 549千字 2018年7月第1版  
印数: 1-3500册 2018年7月河北第1次印刷  
著作权合同登记号 图字: 01-2017-0538号
- 

定价: 99.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147号

---

# 版权声明

© 2016 by Yevgeniy Brikman.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and Posts & Telecom Press, 2018. Authorized translation of the English edition, 2016 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，2016。

简体中文版由人民邮电出版社出版，2018。英文原版的翻译得到 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的所有者——O'Reilly Media, Inc. 的许可。

版权所有，未得书面许可，本书的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：‘如果你在路上遇到岔路口，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*

# 中文版推荐序一

这本书的文稿躺在我邮箱里好几天没有去看，主要原因是个人不太喜欢这个书名。在我的想象中，这又是一本攒出来再蹭蹭娱乐节目热点的平庸之作。但是当我想起来，真正打开书稿开始阅读的时候，我惊讶地发现跟我想的完全不一样。

对，这就是想象世界和真实世界的差别。

程序员或者泛产品技术群体，是非常善于构建一个想象世界的，这也就是第一行代码为什么要写“hello, world”。但是当想象世界需要在真实世界落地的时候，就会产生很多的冲突和不知所措。

因为在写代码的时候，结果都是非常确定性的，编译成就是成，不成就是报警。而现实的创业生活中，有太多的灰色地带，你分不清楚对错，这时候就特别需要有人能告诉你，他当时遇到这样的问题，是怎么做的。

在和脱不花、罗胖一起创立罗辑思维和得到的四年时间里，我们已经记不清有多少次三个人面面相觑，慨叹“创业进入深水区，不会玩了”。然后下一次，我们会发现，上一次的深水区，压根儿才到脚脖子，这次才是真正淹到了脖子。

我们所能做的，就是遍访身边的创业者，问问他们遇到同样的问题怎么解决，过程苦不堪言。而当我看到这本书的时候，眼前不由得一亮，这其实就是把其他人走过的坑，变成你的活地图。如果这本书能早两年出来，就能让我避免走很多弯路。

在想象的世界里，因为你根本没有接触过这个世界的其他方面，很多技术型创业者连沙盘推演都不知道到底有多少“蓝军”。我看到很多优秀的技术人员，在创业中，没有在技术上失败，却败在了产品上、市场上、财务上、税务上、团队管理上……简直就是九死一生。

这本书肯定无法解决很多创业中的实际问题，但是它提供的很多方法论，是可以帮助年轻的技术创业者提前预见可能会遇到的问题的，要么跳过坑，要么换条路。

创业是件很痛快也很痛的事情，上路不易，带张地图吧。

——快刀青衣，罗辑思维 & 得到联合创始人

---

# 中文版推荐序二

## 奔跑的命运

《奔跑吧，程序员》这个名字，让我想到一部电影，《罗拉快跑》。

在《罗拉快跑》里，罗拉需要在 20 分钟内弄到 10 万马克，否则男友就会死于非命。电影用平行结构，展示了三个不同的结局，让观众看到罗拉和男友三种可能的命运。

身为观众，我们当然可以看到三种结局，但如果我们就是罗拉，我们只能有一种宿命。

创业也是这样。创业公司成百上千，命运也成百上千。我们当然可以艳羡那些成功创业者，但如果自己创业，如何才能把握自己的命运？

《奔跑吧，程序员》虽然不能保证皆大欢喜的创业结局，起码可以避免出现悲剧。全书主要从三个方面展开。第一，产品。怎样选择合适的点子，怎样把它打造成产品，怎样验证产品的有效性和价值。第二，技术。怎样选择技术栈，怎样设计架构，怎样交付。第三，团队。怎样组建和激励团队，怎样找到靠谱的创业团队，怎样持续学习。

与通常泛泛而谈的“技术创业”书不同，《奔跑吧，程序员》对每个方面都给出了足够接地气的建议。举个简单的例子，“远程办公”的利弊已经有无数讨论了，但大多数讨论都着眼于形式。《奔跑吧，程序员》直截了当地指出：真正的问题不在于“远程办公”。即便是集中办公，两三个人在咖啡机旁讨论完，并不把过程和结果同步给需要知道的所有人，这种配合方式还不如“远程办公”。

然后作者进一步指出，无论“远程”还是“集中”，要保证效率和健康，团队都应当以“分布式”的方式协作，电子化、公开化、不加锁，同时要确保大家交流不能“就事论事”，一定要有“聊出某些新鲜点子”的机会。这道理看似简单，但仍然有许多创业团队不明白，在这类问题上犯了想当然的错误，结果创业失败，甚为可惜。

《罗拉快跑》的一个结局是，罗拉在赌场赢了钱，男友曼尼找回了钱袋，这个结局堪称“皆大欢喜”。我衷心希望，《奔跑吧，程序员》的读者在创业路上能找到皆大欢喜的结局。

——余晟，“余晟以为”微信公众号主人

---

# 中文版推荐序三

## 创业是人生最好的修炼

近期发生的中美贸易战，暴露了中国企业在基础研发和创新方面的巨大短板。中国人与美国人相比，最欠缺的就是创造力。我说的是那种能够落地的创造力，而不是天马行空纯空想的创造力。

对于中国最有创造力的群体之一——程序员来说，创业是激发自己创造力的最佳途径。大数据、物联网、人工智能、区块链等技术的发展和普及，让现在成为程序员创业的黄金时代。而且对于中国程序员来说，现在是难得可以与美国程序员同步竞争的黄金时代。中国企业甚至有可能在上述这些领域后来居上。

《奔跑吧，程序员》这本书恰逢其时，正是一本专门面向程序员的创业指导书。虽然曾经创业过多次，这本书仍然带给我相见恨晚的感觉，读后受益匪浅。如果你有志于走上创业道路，一定要提前读一下这本书。创业是一件非常艰难的事情，需要具备很多方面的能力。如果你没有做好准备就贸然上路，必然会劳而无功，白白浪费大量时间和金钱。

歌曲《在人间》中有这样一句歌词：“在人间，有谁活着不像是一场炼狱。”确实，人生就是一场巨大的磨难。非常幸运的是，我们是程序员，我们其实有很多创业的机会。人生在世，至少要有一次创业的经验，才可以说不枉此生了。我们可以把创业作为人生最好的修炼，创业的经验可以让我们变得真正强大起来。

——李锐，上海霓风网络科技有限公司 CEO



---

# 中文版推荐语

这本书是程序员技能扩展的好书，在代码上有所建树的程序员很多会自然而然追求更高的挑战：如何带领一支队伍打造一款不错的产品。这本书对想独立创业的程序员来说更具有参考意义。当然我是建议程序员们都读读这本书，至少能知道代码之外原来还有这么多可以去思考琢磨的东西。

——余弦，慢雾科技联合创始人

技术型创业不仅仅需要勇气，还需要有正确的方法。本书通过产品、技术和团队三个方面的经验介绍，让尝试创业的程序员可以少走一些弯路。作者提出的“伟大的公司是进化而来的”观点，值得我们这些程序员创业者去深刻体会和理解，能够帮助我们穿透迷雾找到正确的企业发展方向。

——李启雷，趣链科技联合创始人兼首席技术官

如果你正在寻找一本科技创业权威指南或想深入理解科技创业，那么本书无疑是你最佳的选择之一。作者通过产品、技术和团队这三个维度，全方位系统性地阐述了科技创业方法论。无论你是投资人、创始人、高管，抑或是项目经理、开发者和大学生，这本书都将对你的职业生涯有所裨益。

——彭靖田，谷歌机器学习开发专家，《深入理解 TensorFlow》作者

结合自己的两次创业经历，我深知创业就是一个不断踩坑和挖坑的过程。虽然始终免不了要犯错，但哪怕少一个错误，对基础薄弱的创业公司的存活都是至关重要的。这本书提到了很多避免犯错的技巧，值得每一个创业者学习。

——徐谦，贝米钱包 CTO

创业圈流行一句玩笑：就缺一个程序员了。那我们程序员创业缺什么呢？从构建团队到产品运营，这些技术之外的能力都需要我们一点点补全。而这本书就像一位创业教练，会手把手教你实战，从而极大提高你成功的概率。电子世界的魔法师们，共产主义要靠你我的奋斗，去创造属于自己的时代传奇吧！

——袁行远，彩云科技 CEO，彩云天气与彩云小译程序员

---

# 对本书的赞誉

对创业技术团队而言，良好的计算机科学教育与所谓的“常识”之间存在着巨大的鸿沟。大多数人都必须通过博客、工作，当然还包括学校的磨练才能学到这些知识，本书涵盖了许许多多这样的“至理名言”。我真希望自己在入行的时候就能遇到这样一本书。

——Jay Kreps, Confluent 公司 CEO

作为创业者，你肯定希望只花一点时间，就能学到纷繁复杂的多个学科的大量知识。虽然我很珍惜创业时不断碰壁又突出重围的经历，但还是希望自己开始时就能有这样一本指南。

——Boweï Gai, CardMunch 创始人、CEO

Jim 以广阔的视角带你了解软件创业需要掌握的方方面面。本书不是由术语堆砌而成的，也不是空洞之谈，只是实实在在、简单且被证明可行的建议——这就是我读本书初稿时的直观印象。如果你曾有过“我该如何想出创业点子”“我应该在这个项目中用什么技术”或者“我怎么被那些了不起的创业公司录用”这样的疑问，本书无疑就是为你准备的。

——Eugene Mirkin, Array 风投公司企业家

和 Jim 一样，我的职业生涯也是从大公司开始的。现在我是自己的公司 prismic.io 的联合创始人，我每天都能从这段经历中学到很多。本书正是把许许多多这样的知识呈现给大家。它不但阐明了如何通过创业去释放自己的真正潜力，更告诉你这样做的原因所在。

——Sadek Drobi, prismic.io 联合创始人

如果所有的计算机系都把本书作为毕业礼物送给学生，科技行业将会发生两件好事：最糟糕的科技公司将关门歇业，出色的那些则会明显变得更好。

——Brent Vince, Adacio 公司创始人

本书可谓是无二的一本介绍如何创业的实践之书，阅读体验也非常好。我真希望自己当初踏上创业之路时也能有这样一本宝典。

——Sean Ammirati, Birchmere 风投公司合伙人

献给妈妈、爸爸、Lyalya 和 Molly。

# 前言

```
main( ) {  
    printf("Hello, World");  
}
```

我们在学习一门全新的编程语言时，最先看到的往往是一份“Hello, World”教程，教你如何把文本“Hello, World”显示到屏幕上，由此了解怎样让一个基本程序运转起来。而本书就像一本给创业者的“Hello, World”教程，将告诉你如何打造产品、开发技术和组建团队的。

我真希望自己在上大学时就能读到这样一本书，因为我虽然获得了学士和硕士学位，也有过不少实习机会，但对自己所做的事还完全没有概念。

我自己早期做过一些大项目，比如在汤姆森金融公司（Thomson Financial）开发过一个用于性能测试的桌面应用程序。那时我根本不知道怎么做用户界面，所以就随便把一些文本框、菜单和按钮放到界面上；对如何处理性能问题更是一头雾水，也就是在代码中随意加上一些缓存和线程池；同样，我也不懂得要考虑代码的可维护性，根本就沒去操心测试和文档的事情，反倒是把几千行代码都塞入一个巨大的文件中。

我在 TripAdvisor<sup>1</sup> 的第一个项目则是为一个网页添加一些新的选项，该网页可以列出一个城市的所有酒店。这只是一个简单的任务，就是公司为了让我熟悉一下代码库而已。我在第一个星期就完成了任务，把网页推送到生产环境中。过了没多久，我就被叫到经理办公室，和经理来了个一对一的面谈。我看着他在网页上点开巴黎的酒店列表，选中我加进去的那个新选项，然后就开始等啊等啊，那个页面差不多花了“两个小时”才加载完。好吧，实际可能就是两分钟不到，但我真真切切地感受到，在狭义相对论的作用下，当一个人大汗淋漓恨不得找个地洞钻下去时，时间肯定发生了膨胀。那天晚上，我一直到深夜才发现那段花哨的新代码在进行排序时，每次比对酒店都要调用两次数据库，所以如果要对  $n$  项进行排序，大概需要进行  $O(n \log n)$  次比较。巴黎有将近 2000 家酒店，页面加载一次可能会引发将近 40 000 次数据库调用。那天我虽然没有钻到地底下，但我们的数据库服务器可能差不多要累趴下了。

---

注 1：TripAdvisor 是一家国际性旅游评论网站，提供饭店、景点、餐厅等世界各地相关旅游资讯，官方中文名为“猫途鹰”。——译者注

我不会忘记那段时间出现了多少烦人的问题、丑陋的代码和难看的用户界面，也不会忘记，我经历了多少次网站宕机和多少个不眠的夜晚。但是，最让我耿耿于怀的却是问题这样多，我还找不到答案。比如，我该学习、使用什么技术？为什么我还要费心去考虑自动化测试？怎样才能做出一个不惹人厌的产品？怎样才能让别人使用我的产品？面对工作机会时，我该怎么进行谈判？是要争取更多的薪水还是更多的股权？股权究竟又是什么？我是应该在大公司工作，还是该加入创业公司？

对于上述问题以及其他种种问题，我费了很大的力气才找到了答案。所以，我也尝试着把自己学习到的东西（大部分都是经历了痛苦的磨难和犯了错误之后才得到的）用博客记录下来，或者通过演讲和他人分享。但在意识到有成千上万的开发者也会有同样反复试错的经验之后，我觉得是时候做一些更实质性的事情了。于是，也就有了这本书。当然，有些事情是自己犯过错才能学到的，但除此以外，我希望本书可以让读者从他人的错误中吸取经验，避免重蹈覆辙。

我觉得自己犯过最大的错，就是在职业生涯的早期对创业缺少关注。我的前几份工作都是在知名的大机构（思科、汤姆森金融、康奈尔大学），后来才偶然跳到创业公司（LinkedIn、TripAdvisor）。结果，在这些公司的所见所闻让我惊讶不已。我在创业公司前几个月学到的东西，比之前工作、实习和在学校的那些年加起来还要多。

创业公司并不仅仅是大公司的简化版，就像量子力学并不只是经典力学的简化版那样简单。经典力学描述了宏观物体（比如棒球或星球）以可预见和确定的规则在相对低速的运动中表现出的行为。与之类似，大公司通常都生存在具有确定规则的环境中，它们行动缓慢，因为它们面对的客户和产品都是可知的。量子力学则描述了微观粒子（比如光子和电子）基于某些不可预知、非确定的规则在极高速的状态下运动时所表现出来的行为。同样，创业公司经常在一无所知而又变化莫测的环境中高速运转。虽然很多人都了解经典力学和大公司，但只有对量子力学和创业公司也同样了解，才能看清世界的全貌。由此可见，我们必须从全新的角度去考虑问题，必须用全新的方法去工作。

我们工作的方式很大程度上其实也就是生活的方式，因为我们有一半的清醒时间是用来工作的。难道你不希望用这些时间去做一些可以让自己快乐的事情吗？我曾经认为，从事软件相关工作的人们面对的都是一望无际的小隔间、无能的老板、一份份 TPS（测试过程说明书）报告和各種企业代码。幸好，世界还有另外一面，这就是本书要向你展示的。我会介绍当今世上最出色的创业公司是如何工作的，让你了解这另外的一面。我相信，即便你从来没想过要加入创业公司，这些公司的理念对你也是有所裨益的。随着创业越来越普遍<sup>2</sup>，这些理念也同样会越来越有用。

本书源于我自身的经历和大量的研究，其中就包括对一些程序员的访谈，他们均来自过去十年最成功的创业公司，比如 Google、Facebook、Twitter、GitHub、Stripe、Instagram、Coursera、Foursquare、Pinterest 和 Typesafe（受访者的完整名单见“访谈”）。纵览本书，到处都闪烁着这些人的故事和思想。他们向我们描绘了创业生活的原始景象，不掺杂任何自我营销和公关——单纯就是程序员们在分享自己获得的成功和犯下的错误，并给出他们的建议。

---

注 2：仅仅美国一地，每个月就诞生将近 50 万家小型企业，它们提供了超过 66% 的新就业岗位。

# 本书的内容

我为这本书定下的目标是为创业公司打造一份既实用又具有可操作性的入门指南。这本书包含三部分内容：产品、技术和团队。接下来，我会列出每一部分都包含哪些章，并概括介绍每章具体讲解的技术、工具和技巧。

## 第一部分：产品

### 第1章 为何创业

为何打造面向大众的产品，如今（超越了历史上的任何时候）唯有创业才能为我们创造最佳的机会；什么是创业公司；是什么原因让你选择在创业公司工作，又是什么原因使你放弃在创业公司工作。

### 第2章 创业点子

如何想出创业点子；介绍点子日记、约束条件和痛点的概念；想法和执行的对比；博伊德迭代法则；如何利用客户开发过程快速、低成本地验证自己的想法。

### 第3章 产品设计

介绍大家都应当掌握的设计技能；用户界面应当如何设计才不会让用户觉得自己愚蠢；以用户为中心的设计原则（人物角色、情感设计、简单、可用性测试）；视觉设计的原则（文案、设计重用、布局、排版、对比与重复、颜色）；介绍如何设计最简可行产品（MVP）。

### 第4章 数据与营销

介绍每个创业公司都应当采用的度量指标；数据驱动的产品开发方法；A/B 测试；为什么最出色的产品未必能够胜出；推广、病毒式增长以及创业公司的销售策略。

## 第二部分：技术

### 第5章 技术栈的选择

应该利用内部资源开发软件，还是购买商业产品或使用开源产品；最初的技术栈应该如何选择；如何进化技术栈和重写代码；如何评估编程语言、框架和数据库。

### 第6章 整洁的代码

为什么说程序员的工作并不是写代码，而是理解代码；为什么代码的编排布局、命名、错误处理、不写重复代码原则（DRY）、单一职责原则（SRP）、松耦合、高内聚等可以让代码更容易理解；函数式编程为什么可以让代码易于重用；为什么重构对于编写良好的代码必不可少。

### 第7章 可扩展性

创业公司应当如何调整，才能适应更多的用户和开发人员；怎样修改代码才无须担惊受怕；如何应用测试驱动开发（TDD）获得更好的代码；如何在创业公司中引入设计评审、结对编程和代码评审；为什么说明文档是代码库中最重要的文档；如果你无法测量，就无法解决；如何利用估算推断性能状况。

## 第8章 软件交付

编写完代码之后要做的事情；为什么要使用源代码控制、开源构建系统和持续集成；如何进行配置管理、自动化部署和持续交付；如何为代码增加日志、监控和警告功能。

# 第三部分：团队

## 第9章 创业文化

为何要明确公司的使命和价值；管理层级架构和组织扁平化之间的衡量取舍；公司文化在人员招聘、晋升和激励中的作用；如何为程序员设计理想的办公室；远程办公的衡量取舍；创业公司的沟通策略和方法。

## 第10章 求职之路

如何利用人脉找到在创业公司工作的机会；如何让简历受到关注；如何才能在面试中有出色的表现；如何才能做好白板编程；如何才能提出好问题；在薪水和股权的问题上应该如何考虑；面对工作机会，应该如何谈判。

## 第11章 招兵买马

为什么说人才是创业公司最为重要的因素；创业公司要招什么样的人（合伙人、早期员工、通才和专才）；如何找到出色的候选人（以及如何打造公司品牌去吸引人才）；白板编程为什么是一种糟糕的面试方法（应当采取什么替代方法）；如何给出让人无法拒绝的录用条件。

## 第12章 学习

世界上最引人关注的软件开发者；为什么要撰写博客、文章、论文和图书；为什么要在小组会议、技术演讲和学术会议上发言；为什么几乎所有的代码都应该开源；为什么应该分享自己所知的几乎所有东西。

# 重要观点

除了上面所说的技术、工具和技巧之外，有三个重要的观点会在整本书中不断地出现，它们都是成功创业所不可或缺的，这三个观点分别是：创业与人密不可分、伟大的公司是进化而来的、速度制胜。

## 创业与人密不可分

工作面临的主要问题与其说与技术有关，不如说本质上属于社会学的范畴。

——Tom Demarco、Timothy Lister,《人件》

我们在课堂和书本上学到的有关创业的大部分内容，比如营销计划、产品设计、系统设计、测试策略、招聘计划和组织结构等，其实只是创业过程中的必然产物。仅仅研究这些产物，无法对创业有充分的理解。这就好比被铁链锁在柏拉图洞穴中的囚犯<sup>3</sup>，仅仅研究

注3：这是柏拉图在《理想国》中设计的一个洞穴寓言，暗喻未受过真正教育、不愿面对真实世界的人。

——译者注

面前墙上的影子，根本没有办法完全理解外面的世界是何种景象。

希望本书能让你走出洞穴，不仅了解创业的产物，也能了解创造这些产物的人；不仅学到如何设计出伟大的产品，也学到如何设计出以人为本的产品；不仅学到如何编写有效的自动化测试，也知道为什么有了自动化测试，就可以无须在修改代码时担惊受怕；不仅可以学到伟大的公司是如何组织的，更能学到为什么打造一家伟大公司最重要的是要懂得如何去发现和激励正确的人。

## 伟大的公司是进化而来的

有效的复杂系统一定是从有效的简单系统进化而来的。

—— John Gall

看到长颈鹿的脖子时，你要知道这么长的脖子并不是老天爷一开始就故意设计的。随机突变导致一些长颈鹿的脖子变长，这又恰好提高了它们在某种特定环境下的生存概率，所以成千上万代之后，长颈鹿的脖子就变得越长越长。同样的道理，当我们见到一家成功的公司，必须认识到它的成功并不是创始人在建立公司的时候就计划好的，大多数创业公司都要历经数千次尝试才能有所改变并成长起来，归根结底不过是其中的一些尝试恰好提高了公司在特定市场中的生存概率，只是结果让人感觉是创始人在最早的时候就有所设计一般。

本书将会关注如何以一种增量、迭代式的发展方式去打造一家创业公司（好比敏捷和精益开发），而不是费心去找出完美的计划（好比“瀑布式开发”）。不管你是在打造产品、开发技术抑或建立团队，都会发现最好的起步方式其实就是先做出一个大概可以工作的最小的东西（最简可行产品，简称 MVP），然后再根据客户的反馈（对产品而言）、代码的评审和测试情况（对技术而言）或员工情况（对团队而言）逐渐进化，扩大规模。

## 速度制胜

世界正在快速变化，不再是以大胜小，而是以快胜慢。

——默多克

如果伟大的公司是不断进化且迭代发展的结果，那么迭代得最快的公司终将胜出。所以，本书的很多观点都和如何实现更快的迭代密不可分，也就是如何缩短反馈回路，加快学习步伐，进而提高进化的速度。客户开发有助于更快地发现合适的产品或市场；整洁的代码和自动化测试有助于更快地实现技术；强有力的文化则有助于更快地建立团队。另外，还有一个观点稍稍有悖于直觉，在后面你会了解到，做得更快，完成的质量会更好。所以说，速度致胜。

## 这是一本涉及广泛的书

本书所涉及的每个主题都已经有人写过书，甚至有多本书，所以我们只会关注最基本的概念，让你开启一段“Hello, World”式的学习体验。我们会推荐一些参考资料供你进一步学习。需要说明的是，本书并未涉及创业公司的法律和财务方面的内容。如果你对编



写商业计划、融资和上市等细节感兴趣，可以上网查阅相关书单。

毋庸置疑的是，仅靠阅读一本创业类图书是无法让你成为一名出色的开发者或公司创始人的，就好比阅读健身的书并不能让你成为出色的举重运动员。健身的书可以教会你一些特定的动作和练习，但在你第一次走到杠铃面前时，你仍然只是一只软脚蟹。只有花上无数的时间到健身房里锻炼、出汗，再应用从中学到的知识，你才有可能举得起沉重的杠铃。同样的道理，本书的目的就是要教给你一些在创业中真正用得上的工具和技术，但也只有投入大量的时间去实践这些技术，你才有可能做到得心应手。

即便如此，书中所讲的东西也是没有对错之分的。有人说过，所有的模型都是错误的，但其中有一部分是有用的。书中所介绍的工具和技术在过去已经被证明对创业是行之有效的，我也希望它们以后对你也同样有用。但我们不要把这些东西当作现成的解决方案，而是要在自己的脑海中形成一系列的知识点，在思考问题的时候可以借此得到自己的解决方案。

## 本书面向的读者

如果你在创业公司中工作或者正打算投入创业大潮，抑或就职于大公司，却希望像创业公司一样去运作它，你应该读读这本书。本书将向你介绍如何在瞬息万变、科技创投变化无常的环境下，打造一家成功的公司（或一份成功的事业）所要掌握的所有基本概念。尽管这追根溯底是一本程序员写给程序员的书，但其实只有第二部分“技术”是明显偏技术的，第一部分“产品”和第三部分“团队”能被所有受众很好地理解。

如果你是一名刚入行的程序员，这就是一本适合你的书。本书可以说汇集了我在读大学时所有想知道的东西，也是我在刚开始工作时希望有人告诉我的建议、提示或诀窍。作为一名年轻的程序员，你应该已经掌握了两三门编程语言，也可能精通了几种库和框架，或者在学校里已经做过几个小应用。虽然你还没有做好什么准备，就已经有人会为你的这些技能买单，但你也很快就会知道自己在学校里学到的东西到底能不能很好地用在现实世界中。还是让我先告诉你答案吧：尚不够！那是一条艰难的道路，你也会重复之前的程序员所犯下的无数错误。当然，你也可以好好读读这本书，从第一天起就让自己的职业朝着正确的方向发展。

如果你是一名经验丰富的开发者，本书会让你对每天都在从事的工作有系统性的认识。你还在要求面试者在白板上遍历二叉树吗？在为自己最近的项目挑选技术时，依靠的还是自己的直觉或网上的最新热门趋势？你的待办清单上是不是有一项是“编写文档”？你是不是觉得自己的公司已经变得过于臃肿、行动太慢、缺乏创新？我相信书中的故事可以让你频频点头微笑，你也能够把部分建议应用到现有的工作中，也可能让你决定是时候做些改变了。

如果你是科技公司的经理、执行官或投资人，这本书将帮助你弄明白为什么有时候对时间的估算会和实际情况相差一个数量级，为什么手底下最好的开发者非要跳槽到另一家公司，又为什么最新的“敏捷极限结对XXX”方法并没有让你的团队变得更加高效。你的成功在很大程度上取决于能否理解程序员的思维方式、弄清楚他们一天到晚到底在做什么，以及如何去激励他们。书中的故事可以说都是大白话，哪怕你和下属进行一对一的交谈，也是听不到这些话的。