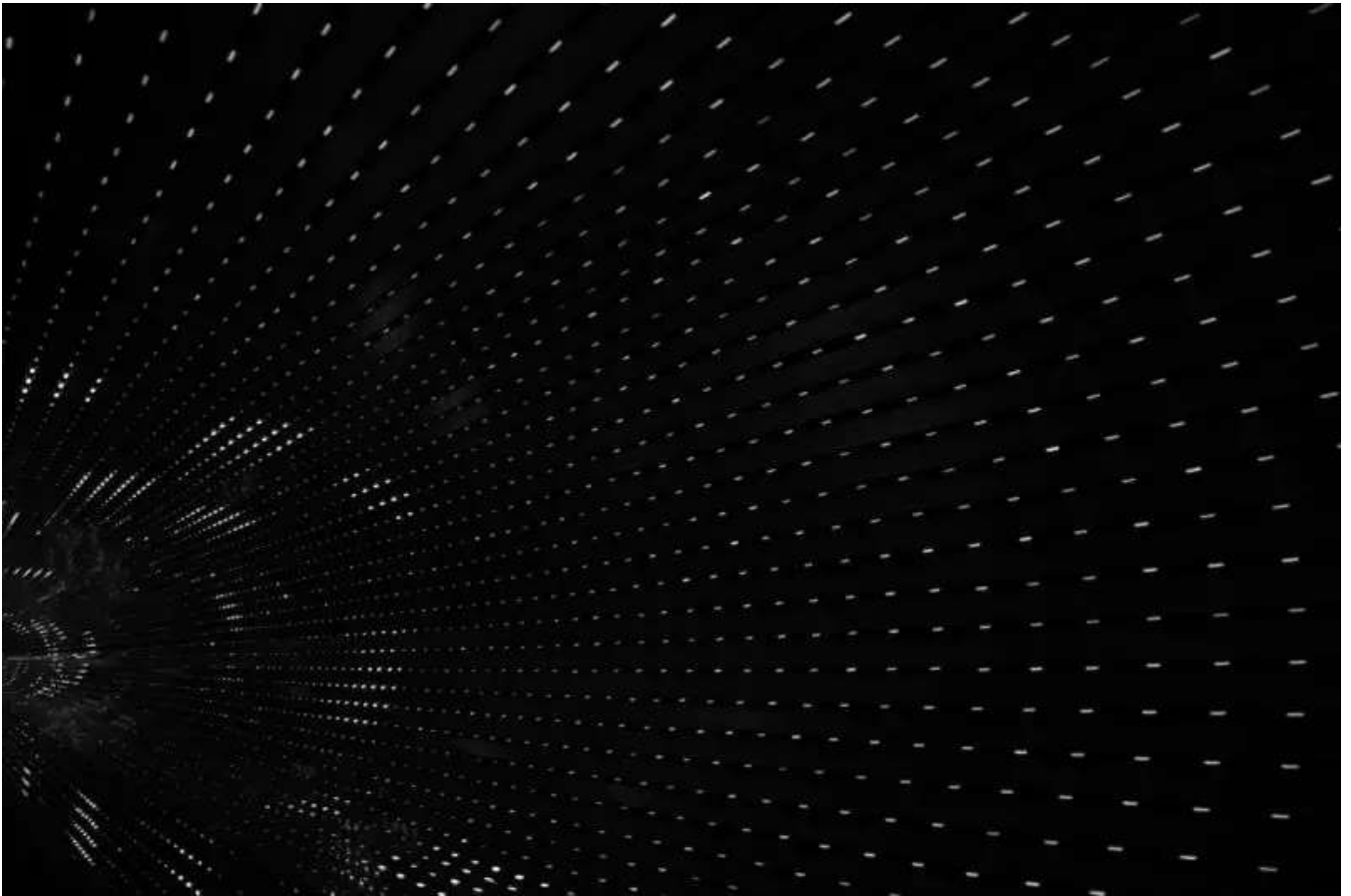


好的技术领导人顶得上100个工程师！

原创：文摘菌 大数据文摘 今天



大数据文摘作品

编译：Zhifu、张文静、王一丁、惊蛰、夏雅薇

一个优秀的技术领导不仅可以帮工程师的工作效率提高10倍，还能够把自身的能量带给团队里每一个成员。那些幸运的工程师们在这些技术领导者的指导下不仅会发现自己事半功倍，而且可以体验前所未有的系统化支持。

对你来说，工作是什么？是养家糊口的工具，还是享受人生的方式？

而对于大多数工程师来说，想要把工作作为享受人生的方式，你需要有一个优秀的技术领导者。

技术领导者会挡掉所有无意义的“工作”，大大加快在有意义工作上的工作效率，从此整个团队会变得无比和谐。

下面这篇文章来自于一名就职于Webflow的工程经理所撰写的技术指导手册，主要内容包括：

- 什么叫做成功的技术领导者
- 当从纯粹的技术角色转型成混合管理和技术专长角色时外界对你的期望，以及如何管理这些期望
- 如何缓解一些常见的问题和挑战
- Webflow在管理上采取的办法

这本手册可以帮助所有勇于挑战自我的领导在这样一个把快乐与成功对立的世界里享受对激情和成功的追求。

技术指导手册链接：

https://github.com/webflow/leadership/blob/master/tech_lead.md

技术领导者到底是做什么的？

根据其最基本的定义，技术主管“对项目的成功负有全部的责任，花费30%的时间编写代码，并利用剩余的70%的时间管理项目”。技术领导者的目的就是指导一个才华洋溢的工程师团队穿过波涛汹涌的水域，引导他们远离危险，帮助他们清除障碍，并让他们充分享受有意义的工作。

然而，说起来容易做起来难。

技术领导者需要无数的技能，不过最重要的几个是真诚的同理心、清晰的沟通和卓越的技术。技术领导者是一个“混合”角色，一方面关注于管理，另一方面关注于工程，并且他们也是项目期望和任务开发之间的联络人。一个项目的成功取决于技术领导者，也取决于公司是否能够确保他们充分获得所需的支持。

为什么我想成为一个技术领导者？

相比一般的工程师来说，技术领导者承受了更大的压力，而且在平衡团队管理和贡献代码的需求上是非常具有挑战性的，特别是首次进入领导者角色的时候（顺便说一句，这很正

常)。

也就是说，管理者的生活非常有意义，他们在决策链上有更高地位，对于Webflow用户的影响是至关重要的。在这个职位上，你会发展你的影响力，这提供了更多职业发展机会。这个角色经常被视为“高级”工程师的进阶，也是工程经理职位的必要条件。你将指导并帮助其他工程师成长。有些人很享受这种新挑战以推动他们个人到达新的高度。

我如何成为技术领导者？

尽管去问！是的，就是这么简单。向你的经理一对一地表明你有兴趣成为技术主管。你的经理有责任为新角色设计一条路径，并且根据你目前的经验确定是否将你指定为下一个项目的技术主管——如果不是，则为你提供成为技术领导者所需技能的培训机会。

当我管理团队的时候外界对我有什么期望？

技术领导者的工作包括如下的责任（排名不分先后）：

- 与产品经理紧密合作，在最后期限前后设置合理的期望，并明确在什么情况下，项目被定义为难以按时完成
- 将项目分解成可完成的小任务，并将任务进展设置成迭代模式，保持跟进阶段性进展
- 为团队提供充足成块的工作时间，以便让他们能够经常进入专心致志的工作状态，并充当团队的门卫，阻挡其他事去让他们分心
- 确保你的团队在任何时候都有足够的工作，避免团队成员处于想要工作却无处可去的状态
- 多做代码审查，第一个去做质量检查，并在必要时候自己也可以写代码
- 在团队成员执行任务的时候尽量给他们机会与自己交流（合理估计有多少时间自己可以专心工作，有多少时间需要留给团队做交流）
- 时不时与其他部门开展合作

案例链接：

https://github.com/webflow/leadership/blob/master/tech_lead.md#help-we-are-behind-schedule

技术主管可以管理哪些不同类型的团队？

Webflow将其有才华的工程师安排到一个技术主管负责的（Action）和Permanent团队。

行动团队：在一个任务下组成，任务完成时解散

永久团队：主攻一个主要问题且永不解散。如绩效与稳定团队。技术主管和队员可以在这些团队中轮岗。

我应该期望什么规模的团队？

每个团队规模不尽相同（甚至可以有一个联盟那么多），但一般来说，团队包括三名成员，其中包括技术主管。管理两个人的团队相对容易，但是一旦管理三个及以上的团队，交流的可行性就会急剧下降。这并不仅仅是孤立技术主管的关系，而且也 and 团队成员之间如何相互沟通有关。较大的团队当然是可以工作的，但三人团队是一个更好的起点。

这并不是说一个团队只能有三名成员。一个行动小组可能包含七名成员，包括一名技术主管。技术主管可以将团队分成两组，并将每个小组的重点放在相似功能范围的并行任务上。然后由技术主管决定每个团队的团队主管，并让他们负责他们所在的团队工作。请记住，每个小组都应该关注功能效率并彼此合作来解决问题，从而减少由于并行工作而经常产生的延迟。

前述的团队结构可以由后端和前端工程师组成。Webflow希望模糊这些工程学科以及非工程学科之间的界限，例如设计师。形成跨学科团队是功能效率的目标，无论是你个人的要求或项目的要求。

团队效率和团队管理

团队设计有两种方式。第一个是以“功能”效率来设计团队，这个方法有利于团队合作来解决相关的问题，另一个方式是以“资源”效率来设计团队，这种方式利于各成员来同时解

决不同的问题。两者都有自己的优势，但我们要求技术主管尽可能地优化这两种方式。

专业提示：并行化需要明确的范围。如果你在带领一个迭代设计规范的项目，那么最好只是优化功能效率。

求助！我们的进程已经落后了！

保持平常心，去喝杯咖啡或晒晒太阳，再回到办公桌前，心静才能自然凉。

案例链接：

https://github.com/webflow/leadership/blob/master/tech_lead.md

几乎每个项目都会遇到一些未知问题而影响交付日期。不要试图去回避这种情况，试着去接受。你需要将这些未知问题加到时间估算中，认识到遇到未知问题是很正常的。这是好与非常好的区别。

将没有按时完成任务等同于罪责其实是士气的杀手。士气其实是你团队最宝贵的资源。我们最好将“延后”理解为“进展推后”。Webflow理解软件开发是艰难的，所以我们可以帮助你不能如期交货变为“进行中”的状态。

关于项目管理

在我们深入研究“返工”“延期”“放弃”之前，有两个重要的项目管理概念将帮助你理解为什么要遵循这个期限。

首先，需要强调工作成果与固定日期联系起来的必要性。如果没有明确的目标，进展很难衡量。我们必须衡量一些事情的进展，即使这只是一种猜测。进步是激励的生命线。

其次，你可以通过四个指标来帮助项目回到正轨：

- 时间：工作成果的发时间
- 质量：工作成果的技术含量
- 资源：多少人参与了这项工作
- 范围：这项工作是什么以及是用来做什么的

随着项目的发展，这四个指标可能会发生变化。它们是项目经理有效管理的工具。也就是说，Webflow生产尽可能高质量的产品，但是不会牺牲时间，资源或范围这另外三个杠杆，所以我们只需要使用这三个杠杆，下文会做进一步说明。

专业提示：技术主管通常是满足企业需求的底线。他们的决定必须置于这样的问题中：“如何保持公司健康成长？”。如果你的商业头脑很少或根本没有，你可以看看乔希考夫曼的“个人工商管理”。这是现代商业实践中的精彩课程，可以帮助你在考虑Webflow的需求和团队需求时做出更好的决策。

课程链接：

[https://www.amazon.com/Personal-MBA-Master-Art-Business/dp/1591845572/ref=sr_1_1?s=books&ie=UTF8&qid=1513878441&sr=1-1&keywords=The+Personal+MBA\)](https://www.amazon.com/Personal-MBA-Master-Art-Business/dp/1591845572/ref=sr_1_1?s=books&ie=UTF8&qid=1513878441&sr=1-1&keywords=The+Personal+MBA)

返工/延期/放弃（缓解策略）

在与你的产品经理讨论截止日期可能有变数时，你有三种选择。在这里按照考虑顺序排序如下：

返工

延期

放弃项目

返工

返工应提出两个问题：

- 我们能否为项目增加资源以如期交货？
- 我们能否改变工作的范围以如期交货？

在评估如何避免错过的期限时，质疑资源和范围应该是第一个工具。首先考虑是否有更多的资源可以帮助解决这种情况，但通常情况并非如此，除非该项目最初人手不足。添加后期资源甚至会将截止日期推得更远！所以，你的下一个工具是缩小范围。

推荐阅读：

https://en.wikipedia.org/wiki/The_Mythical_Man-Month

专业提示：缩小范围通常是在尝试如期交货的同时还能提供商业价值的首选。项目需要更多资源才能如期交货的可能性在10%的范围内，90%的可能性为缩小范围。

缩小范围通常是可行的。作为充满激情的软件开发人员，我们倾向于精钻而不是粗放。这是你可以把工作切分成更切合实际的期望的小块，并和利益相关者沟通这些实际的期望的机会。

延期

如果范围不能缩小，并且增加资源也不是一种选择，那么下一个最好的选择就是延后期限。

“那么，如果期限是宽松的，那么期限到底是什么时候呢？”

呃，我们尽全力避免更改期限，但有时还是会发生，这其实是没有问题的。当我们试图达到不切实际的期限时，会有太多的事情处于危险之中，其中包括团队倦怠，产品质量差，士气低落等等。

这里要强调的不是要忽视期限。关键是，如果不能按时交货，项目将会陷入僵局，并且项目会向未知的方向发展。这比延期更糟糕，所以延期吧！

放弃

最后的和最少的选择是完全放弃这个项目。如果你（或其他利益相关方）发现最后的成果会对公司产生负面影响。请考虑废弃这个项目！注重高效工作，而不是生产性工作。

可选的第四种策略：紧盯项目进展

还有第四种选择，即不能按时完成的威胁只不过是内心的警钟，那就需要密切关注这个项目。当你的直觉有威胁时，就要着重关注。走在问题的前面非常重要，而且这时可以先发制人。

专业提示：让大家更轻松的关键是掌握管理期望的艺术。只要你保持坦诚，少说空话，多做实事是明智的。说实话，如实时表示对不能如期交货或失去核心资源的担忧；会比预期更早的完成工作可能性等。你要像你对未知事物持怀疑态度一样诚实。

怎样才能做出更好的估计？

在撰写本文时，没有人发现用于预测软件开发时间表的魔法。其实软件评估很少准确。这是敏捷哲学的核心：迭代和发现，然后交货和改进。这是一种发现的艺术，而不是交付的艺术。

Webflow遵循一个迭代过程，如其他部分所述，预估是重要的，但没有揭示未知问题那么重要。

迭代过程链接：

https://github.com/webflow/leadership/blob/master/tech_lead.md

这里有一些帮助准确预估的策略：

预估未知

发展很少按计划展开。你可以给任务最佳猜测，但不是精准预估。把最佳猜测的时间乘以四倍，特别是如果该任务涉及未知问题。这听起来有点疯狂，但是有时候是需要这样的。当然经验有助于技术主管改进这个方程式，但这是一个很好的起点，为未知数留下解决空间。

叠加未知

一旦你创建了你的主要追踪问题，你可以大概了解项目可能需要多长时间。一定要确定哪些任务与发现未知问题相关，哪些具有更多具体的方案。发现所有未知问题后，你将对交货日期的准确性有更好的理解。

二八原则

我们很容易忽略时间消耗的细微差别，从而减缓项目的最后20%的完成。当全面地查看项目时，请使用80/20规则将其分解。项目的最后20%可能占整个时间轴的80%。造成这种情况的原因有很多，但最后的20%往往是对工作成果进行完善，完善每一个功能和每一个角落上很复杂的，这些情况在项目结束时会出现重合。

这就意味着只要将项目中的前80%当做还是在途中。这将使预期与项目后续需要的精细度相符。

永远不要忘记测试！

当你预估截止日期时，记得要为代码完成选定一个日期，以便QA有时间来发现错误或未解决问题。预估时间这中必须考虑进这个“检验”时间，同时也应考虑QA目前的工作量。

冷却：错误修复后交付

在交付时，应该计划留出一些时间，在开始新的里程碑之前解决任何即时错误。时间的长短可能因可交付成果的复杂程度而异，一般选择一周为一个时间节点。这是一个让你的团队在进入下一组任务之前休息休息的机会，同时也有助于收紧下一个里程碑的MTI。

项目需要多长时间？

虽然功能的范围可能需要数月 and 数月的工作，但其版本里程碑的目标应该是六周的时间表，包括质量保证，因此每个里程碑的代码都是在四周左右完成的。这样市场营销部门就可以评估一系列功能并将其放进一个个袋子里，并根据市场趋势对它们进行排队。将一个大特征分解为六周的时间线可能会带来挑战，但有几个重要的原因：

- 较小范围和时间表的推理要容易得多
- 如果项目的业务价值微薄，它可以让项目发挥重要作用

- 它允许三人小组效率更快

一个六个月的项目主要里程碑可能看起来像这样：

- Alpha版发布（6周）
- Beta版发布（6周）
- Feature Launch v1.0（6周）
- Feature Launch v1.0.1（6周）

我应不应该分支功能分支？

答案是：不。

不要将功能分支分开。技术领导的目标应该是让他们的团队将他们手里的功能分支直接提交给开发人员，而不是另一个与开发人员保持同步的功能分支。

长寿命的功能分支通常会引入代码依赖性和其他编程模式，这些模式会面临难以保持同步的分支问题。相反的，技术主管应该将他们的项目置于特征标志之后，并不断将其与开发者合并。

总而言之，Webflow有两个主要分支：

- 开发分支
- 主功能分支

其中，特征分支

- 可能来自：dev
- 必须合并回：dev

功能标志

我们鼓励我们所有的工程师每天都推送代码（如果可能的话），并且为了防止新功能让用户不适应，我们建议技术领导将这些新功能后面放置可以切换的“功能标志”，同时放置如“点此获得帮助”的概要性介绍。

我们承认可能有一些功能承诺可以被长期使用。但“可能”也意味着，可能在1%的情况下我们需要重构最基本、最重要的那部分内容。所以，“可能”在某种情况下也意味着“基本不会”。

如果需要这样一个分支机构，请告知整个团队，你的产品经理和你的工程经理你的意图。你可能会惊讶于如何将这项工作组织成规模较小且不断合并的分支机构。

我怎样才能保持“专注”？

保持“专注”意味着你首先关心自己，找到解决问题的“快乐”之处。生活就是要进行尽可能多的有意义的工作，而不仅仅是进行有意义的人类活动。

这意味着在日常工作中，需要时不时地休息一下，并参与让你保持新鲜感和专注的活动。有没有考虑或实施过下面的方法，比如：阅读一本书，看Netflix影片，锻炼，或去外面呼吸新鲜空气。找到一个能让你保持工作和生活点的例行公事，不要害怕向你的经理表达这些需求，也不要害怕为他们腾出时间，即使你可能觉得这些事件正在削减你的生产力。

如果你无法做得到专注，那队伍也很可能不会有凝聚力，你要做到身体力行。

我如何保持有条理性？

新技术领导者感到不知所措，有这样情况的原因往往是，他们可能没有完成某些工作。（当然，我们中的一些人可以跨越这个阶段，但对大多数人来说这却是一个躲不开的经历）。

学习并运用时间管理的艺术有助于减轻压力。具体来讲，可以通过多种方式来实现，并通过经验总结来形成自己的经验体系。

如果你从未阅读过时间管理方面的书，我们建议你先从David Allen的Getting Things Done开始。阅读此书有助于学习如何将头脑中的杂音清理出去。如果他的方法不适合你，那就找别的方法，如果有效，可以把它与他人分享。

我应该被期望写多少代码？

这取决于项目，但一个可行的估计方式是，在整个项目的30%的时间内（或者更少）进行编码，30%的时间（或者更多）检查代码，并用剩余时间为项目团队提供服务。

代码审查

你要对最终可交付成果的质量负责，也因此你需要审核并在每个项目报告上签字。这样的检验机制对于大型团队的工作来说会非常耗时，所以最好鼓励团队进行内附代码审查。也就是说，指导团队成员（无论是初级团队还是高级团队）来进行代码审查，并将指导视为一个让你进一步掌握自己技能的机会。

我如何向All Hands和Lattice提供状态报告？

每周四上午11点（截至撰写本文时），Webflow都会举行一次“全员参与”会议，届时管理团队将传达所有Webflow正在进行的项目以及大型公司目标和举措的状态。技术主管有责任在本次会议之前将其项目的进度更新提供给Webflow项目跟踪器Google文档。文档在Slack的# all-hands频道中共享，更新的模板位于Google文档的末尾。该模板中的项目有：

- TDLR，该项目状况的简要介绍。
- MILESTONE ON-TRACK / OFF-TRACK，你可以为每个活动里程碑提供跟踪报告，其百分比进度以及前一周的百分比变化（这些是猜测），也可以列出团队将在未来两周内完成的任务和预计的交付日期。
- 关键决策，你提到任何导致时间表更改的重大决策，范围更改以及与支持/市场营销或资源更改有关的任何事项。
- 风险，未知和封锁，你可以提上周以来出现的任何风险，未知或阻挠团队前进的东西。

Lattice工具

Webflow使用Lattice工具帮助追踪更高层次的公司目标。除了你的每周All Hands更新外，我们还会要求你对分配给你的Lattice目标进行更新。如果你没有帐户，请联系你的

工程经理。

我如何保持团队的积极性？

提出进步意识，为创造性地解决问题提供足够的空间，而不仅仅是“萝卜加大棒”的激励方式。我们是有主观能动性的生物，很多事情都没那么复杂：将期望达成的目标、工具及方法路径指出来，就会收获结果。

这里是我们流程的一些基本机制：

自治，掌握和目的

Daniel Pink在他的书《驱动力》（Drive）中否定了人类是由外部因素（比如金钱或更好的办公室，职位等等外部因素）驱动的说法。相反，他发现我们受内部动机（例如被赋予归属感，提高技能的机会，并且按照我们自己的条件这样做）因素驱动。这三个内在因素可以归结为自治、掌握和目的，并且是解析动机基础的极好起点。

提供这些关键激励因素的职责一部分落在Webflow的肩膀上，但聪明的技术主管也可以使用它们来产生很大的影响。所以，每周都问问自己下面这些问题：

- 我是否给予我的团队足够的空间，让成员用自己的方式解决问题？我是否在应该指代指导和意图时明确地给予了指令？[自治]
- 我是否将团队成员安排在他们可以正确成长的位置上了？[掌控]
- 针对世界上出现的一些问题，我是否对Webflow进行了功能性的调整？[目的]

提供反馈

从哈佛毕业后在谷歌和苹果担任执行官的Kim Scott在她的《Radical Candor》一书（此书还未在国内出版）中就如何最好地管理与团队中每个人的关系和期望进行了陈述总结。

事实证明，我们不应该淡化我们的感受以及我们对彼此的看法，相反，我们应该以充满人文关怀的方式来就问题进行讨论。这项原则的基本前提是“给予关注，直面问题”。这意味

着，你必须让你的团队知道：你理解、体谅他们，同时证明你对团队及团队中个人利益的关心，同时仍向他们提供重要的反馈（虽然可能会使成员不开心或利益受损）。

尽早经常性地提供重要的反馈意见、表示你对团队成员的关心，会使你在未来的道路上避开许多必须面对的灾难性挑战。此外，反馈意见及表示关心不仅适用于消极情况，同样也适用于积极情况。因为无论情况是好是坏都是重要的。

专业提示：我们框架反馈的方式可以改变接收效果。要就事论事，而非就事论人，综合考虑这种结构中的使用场景，表现及影响模型。它的工作原理如下：找到问题发生的场景，将表现及影响单独拿出来处理。例如这样：“在今天的会议中，你多次打断了Brian，让Brian觉得他直到会议结束时才能发言，表述他的想法。这让会议比实际所需要的时间更长。”

流动性

使团队中每位成员都有岗位调整的可能性非常重要。提供潜在的机会本身就足以让大多数人在工作和生活中感到幸福。这是激励和工作效率的关键因素，因此我们在此特别就其进行提醒。

我的团队中有一位表现不佳的成员。我该怎么办？

你有没有听过这句古老的格言：“没有糟糕的员工，只有糟糕的经理？”呃，这句话在大部分情况中是符合实际的。Webflow雇佣了才华横溢的工程师，因此，在你作出任何关于表现欠佳的问题的结论之前，确保你为团队提供了100%的服务。

每个团队成员都必须有足够的动力，有充分的机会为自己创造有意义的进步，拥有自主空间和获得感。因此，提供持续的反馈也是激励团队的一个重要组成部分。

你也必须考虑团队成员的内在工作生活。问如“事情怎么样？工作之外的一切都顺利吗？”之类的问题是没问题的，并且你应该经常这么办，但请记住把握好尺度，不要打探个人隐私。在记住个人问题的同时，给你的团队成员讨论个人问题的空间。

如果你已经尽最大努力为你的团队成员创造适合他们工作的环境，并且他们仍然没有达到你的期望，那就与你的经理聊聊下一步该做什么吧。

我怎样才能避免毁掉我的团队？

如果一个团队无法完成最后期限，这是一个管理问题，而不是团队的问题。这意味着，在这个过程中的某个地方，这个项目没有按计划进行，项目计划也没有得到及时纠正。

因此，避免职业倦怠的第一条规则是“很好地管理项目和期望”。

第二条规则：永远不要在还没考虑好就向团队提问（而且你不能让自己上夜班和周末）。

其他组织可能会要求他们的团队在遇到困难时加班工作。但是团队应该是一个长期性的团队，要注意长期的效率。Webflow深深关注其团队，不仅在专业上，而且在个人方面，所以我们必须尽我们所能来好好管理我们的时间。

关键时刻

哦，关键时刻到了，最好的团队已经得到了，但还有不可避免的问题会出现。

作为一名技术主管，你将不可避免地遇到一个即将到期的截止日期，并且可能需要稍微付出一些努力才能使团队在最后阶段把项目拿出来。所谓的“稍微”意味着你的团队可能需要在每周40小时的工作时间内多花几个小时。

这样的话就对了，我们的“紧缩”版本并不需要在晚上和周末各种加班，通常只在极少数的情况下这么做。一个人一天的高效率工作时间通常只有2-4小时，在此时间之外，很难做到不出错的做好工作。Webflow作为一家公司，要求他们做更我的工作收获的会是严重的收益递减（效率下降），同时也会使工作的人员感受十分糟糕。

加班赶工是真实存在的，但也可能是管理不佳的一个表现。我们必须尽最大努力将这些过度紧张的时期限制在每年一到两次。

推荐书目

Flow

- Deep Work
- Drive
- Leaders Eat Last
- The Manager’s Path
- The Progress Principle
- Getting Things Done
- Getting To Yes
- Radical Candor
- Search Inside Yourself
- Now Discover Your Strengths
- The Personal MBA

相关报道：
<https://hackernoon.com/the-effective-tech-lead-is-a-100x-engineer-fe49c0372a63>

【今日机器学习概念】
Have a Great Definition



志愿者介绍

回复“志愿者”加入我们

