



# 「译」自己写的才是最好的样板

📅 2017.10.30 👤 Scarletsky ☰ technology 🌡️ 热度 44°C

样板应用非常有用，它们可以令你的下一个项目非常快的构建起来，并且提供了目录结构让你有章可循。

然而，它们有时候也会浪费大量的时间。开发者们通常都想知道样板的底层是怎么运行的，他们会浪费时间在研究上。除此之外，样板对你的需求来说要么太精简，要么太臃肿，这意味着你还需要花时间去添加基本的功能，或者删除那些没必要的东西。

显然，这个过程本身并不是一件坏事，但这很容易会导致项目延期。你更应该把时间花在实现功能上！

## 我们应该用样板吗？

当然！我并不是说我们不应该用样板。恰恰相反，相比使用样板，从零开始构建项目消耗的时间会多得多。但你应该为你的项目使用正确的样板。什么样的样板才是正确的样板呢？我认为你自己写的那份才是正确的样板。创建自己的样板并不只是让你更加了解正在使用的技术，它还会为你留下一些你已经了解内部原理的代码片段，并且可以应用到下一个项目当中。

当然，你并不需要从零开始实现。你可以选你觉得有用的样板，然后自定义它来满足你或你的团队的需求。

## 但并不是所有项目都是一样的！

我同意，进一步来讲，不是所有环境、项目团队都是一样的。例如，你可能会在自己的 side projects 中使用和在办公室时完全不同的技术栈。从我个人而言，我写了 4 个样板应用，分别应对不同的项目、同事、客户。举个例子，我写了以下几个样板：

- 一个单页的 VueJS 应用
- 一个提供 API 的 Express 应用
- 一个静态网站生成器(Hexo)
- 等等

关闭

[博客](#) [分类](#) [标签](#) [关于](#) [RSS](#) [搜索](#)

理想情况下，你应该让你的样板应用保持更新，确保你使用的依赖是最新的，并检查它们是否依然正常工作，同时要注意更新时的注意事项。慢慢地，你会知道哪些库工作正常，哪些不正常，可以用哪些库去取代它们。但不要更新得太频繁，否则我们就陷入了不断花时间去修复我们的样板应用的循环中。我们应该花更多的时间在那些能改变世界的项目上！

总之，你需要平衡一切。

## 参考文档

[https://mindthecode.com/the-best-boilerplate-app-is-the-one-you-code-yourself/?utm\\_source=wanqu.co&utm\\_campaign=Wanqu+Daily&utm\\_medium=website](https://mindthecode.com/the-best-boilerplate-app-is-the-one-you-code-yourself/?utm_source=wanqu.co&utm_campaign=Wanqu+Daily&utm_medium=website)



支持一下

