

前端 UI 工程师的困境与破局

这个话题几年前我就一直在思考，曾经我也一度很迷茫，现在虽然已经不做网页重构三年了，但是仍然会有些人问起我的经历和现状，也总有一些现实在刺激着我不得不去想这个问题。同时，这些工作经历和思维方式也一直深深地影响着我，所以即使这终究是个怎么说都会得罪人的话题，我也不得不勇敢面对。

不过事先声明，这篇文章不会给你带来什么现实的答案，毕竟每个人的境况不同，内心的追求与理想也不同，不能一概而论。我只是抛出问题和想法，希望能对你有所帮助。

从一个认知说起

所有从事互联网行业的人，都知道有个工种叫做“美工”，当然他还有许多其他的名字。比如，南方的就喜欢自称“页面仔”，“切图仔”，而 tx 的就喜欢称为“构建”，意思是构建页面结构的工种。虽然美工，构建这些名字看起来没那么高大上，但也不能说带有歧视，只是我们都不太喜欢这些名字。而且，早些年真正的美工的工作，是可以从设计到实现一手包办的（现在称为全栈设计师）。

那么，这个岗位从什么时候开始变得有点 low 了呢？这可能要从《网页重构》这本书，以及这个岗位的诞生开始。重构，就意味着推翻过去的东西，重新构建新的规则制度。美工，由于参与的流程比较长，所以也不可避免地难以做到每个环节都很完美，甚至很糟糕（全栈工程师？呵呵）。所以导致的结果就是，大量素质很一般的从业人员乘着互联网的大潮，进入了美工这个岗位，最后设计设计做不好，代码代码写不好。最终留给人的印象就是：美工都在做一些粗制滥造的东西。

这个结论我只能说，确实有这样的现象，但也不能以偏概全。而且，现在除了有些公司还在招所谓的“淘宝美工”之外，几乎也见不到用美工这个 title 来招聘的了。不过单说写 CSS 这件事，却不只是美工在做的。今年的 The Ultimate CSS Survey 中第二题就是关于角色的调查：

Which of the following most accurately describes you?

- A. I'm a designer that writes CSS
- B. I'm a front-end developer who does some design
- C. I'm strictly a front-end developer
- D. I'm back-end developer who writes CSS occasionally
- E. I'm a full-stack developer

From [The Ultimate CSS Survey](#).

我们可以看出，无论前端后端还是设计师，都可能会写 CSS，或者说都可以写一些 CSS。这也从侧面说明，仅仅是 write CSS 并不能成为核心技能，do some design 也是一样。那么 HTML 就更不用说了，单纯技术层面的两板斧，都不能成为核心竞争力。因此，大家心里一个普遍的认知就是 HTML 和 CSS 很简单，真正的程序员也经常对这些东西嗤之以鼻。总体的印象就是：“简单”，“算不上程序”，“不就是切个图的事儿吗？”。对，这就是对于这个岗位的一个基本认知，至少不了解这个岗位的人，只能联想到这些了。

身份的错位

由于这个角色的 title 实在太多，我还是统一用 UI 工程师比较好，至少是个比较新的 title 吧。既然是新的 title 必然会有些新的定位，以 tx 为例，许多部门的 UI 工程师也都归到了 T 族（技术族）的通道中（曾经有一段时间是在设计族）。然而，尽管职级通道划分到了技术族，但编制往往是在设计部门，或者是跟着产品线走。那么设计部门中的 UI 工程师，就成了设计师中最懂代码，程序员中最懂设计的一拨人了。

这种略微有点错位的身份关系，也曾让我们迷茫。说是设计师吧，也做不出像样的设计稿；说是工程师吧，只做些页面搞搞动画好像也谈不上工程。唯一一个经常被我们挂在嘴边的词就是“用户体验”，但用户体验又是个很大的范畴，我们能把握的也只有页面的性能，酷炫的动画等等。这些东西确实可以从某种程度上提升体验，但究竟有多大作用，很难去量化。

比如一个按钮，加了 CSS3 动画的比没加动画的点击率高，这就一定能说明体验更好了吗？在页面总 PV 没变的情况下，这个动画是否会让这个按钮哗众取宠，影响了其他元素的关注度呢？当然，我们可以用数据说话，用基数足够大的 A/B Test 去印证它的效果。但是实际工作中，真的有资源去支持你验证这些小动画的作用吗？

所谓身份的错位，是你的角色在部门不是很重要，在整个产品研发流程中也没那么重要，你的价值很难体现，或者悲观一点说，就没那么大价值。

为什么不受重视

前面我说到 UI 工程师不受重视，这可能不是普遍现象，而且就我本人的经历来讲，我已经在有限的资源内，得到了足够的重视。我所说的不受重视，其实应该说是推动力有限，在整个产品流程中的影响力有限。比如你对你所做的页面有些不同的想法，尽管希望很渺茫但你还是希望能对你做的产品产生一点影响，那么在 UI 层面，向上游你可能需要说服交互设计师、设计师、产品经理，向下游可能还需要跟开发去沟通。所以在常规的产品研发流程中，你几乎很难去影响到你做的产品。

这里面有个特殊的方向，就是电商类的运营活动和卖场，以及各种游戏宣传页等需要大量应用动画效果的地方。在这样的项目上，UI 工程师可以向需求方提供专业的动画效果建议，但，往往也要说服设计师同意你的效果，并且配合你一起制作图片素材。

再举个性能优化的例子。相信很多 UI 工程师都做过所谓的页面性能优化项目，提升首屏打开速度什么的。进行这样的项目往往要先分析页面性能瓶颈，拿到当前的参考数据，再从文件大小、图片大小、连接数、动画帧数、脚本加载时机，blabla 等等逐项提升。这其中，文件大小可能是开发人员搞定的；图片大小可能需要运营人员和设计师一起配合，如果规范不好，即使自动压缩也很难尽如人意；连接数可能需要运维支持推进服务器 combo，或者前端开发推行本地合并；动画帧数什么的倒是 UI 工程师可以控制的，CSS 文件大小也可以控制，不过很可能运营上传一张大图，你的 CSS 压缩就白费了。从这个例子可以看出，除非你有强大的项目推动能力和跨部门沟通能力，所以类似这种项目，往往都是虎头蛇尾，无疾而终了。

凡事都有两面，不受重视也一样。一方面你可以说你明明很努力了，但是领导就是不认可；另一方面，也可能是你真的没有体现出足够的价值，所以才得不到认可。这其实是个悖论，无法明确因果，很多事情没有领导支持就推行不下去，推不下去就体现不出价值。而另外一些事情呢，领导重视了，期望高了，你反倒可能达不到领导的预期了。这里面具体可能又涉及到复杂的 KPI 导向以及一些人情世故等等，往往不是写代码的 UI 工程师能参透的。所以，这里我们能做的往往是，尽人事，听天命。

分工问题

GitChat

说到底，所有岗位的产生都是有社会分工造成的，而在 UI 层面，许多公司单独分离这样一个工种也是合理的。从个人角度，专精和广博本就很难权衡，设置细分岗位有助于专精。虽然很多人做着做着就得到了瓶颈，但从职业发展的角度，先专精一个方向是对的。从公司角度，职业的细分可以提升组织的稳定性，降低用人成本。一个再优秀的 UI 工程师离职，也不会造成太大的伤害，不可替代性没那么强。更何况，设立这个岗位的公司本就不多，所以一定程度上形成了劳动力的垄断，因为跳出这些公司的圈子，就很难找到工作，所以轻易也不会动。

话说回来，公司的目的是盈利，虽然有基本的社会责任和愿景，但本质还是为了赚钱创造价值。那么在这个基础上的福利，公司文化，岗位设计，职级体系，薪酬体系，都是服务于这一个目标的。所以在 UI 工程师这个岗位的设计上，对公司是没有坏处的。当然，公司足够大的情况下，具体做的事情还是有很大差异，比如有些 UI 工程师要写 native 应用界面，有些只负责 web，还有些要写 JS 和 PHP 等。我说这些的目的，是想告诉你，不要试图从本质上去修正和抱怨 UI 工程师这个角色，因为本质上，它是合理的存在。

焦虑的来源

在这样一种错位的身份下，每一个 UI 工程都不可能不焦虑，即使是已经做到团队 leader 的同学也一样。为什么呢？因为 UI 工程师的群体正在弱化，即使是 UI 工程师本身也并不看好这个岗位。不可否认，确实还是有几家大公司里，有那么一群拿着很高薪水的 UI 工程师，但是在大部分情况下，他们也离不开这几家公司了。因为，能提供这个岗位，或者说对 UI 层面有这么高要求的公司并不多了。

管理者也是一样，管理 UI 工程师的经验未必能去管理其他前端工程师。而从 UI 工程师做起能够一路攀升的更是凤毛麟角。这真的是能力的问题吗？当然不是，也没有人愿意承认。只是这个角色限制你只能看到那一点天地，角色限制了你能接触到的人员，也限制了你的思维方式，和影响力范围。

记得我做页面重构的时候，我可以影响产品去改改 UI，可以帮运营去优化工具，可以给开发提建议去优化代码。但是，作为普通员工的我，几乎没有什么机会见到更高级的领导，也没有什么需要我们这个角色露脸汇报的。而开发和运营就不是，往往一个重点大项目就需要跟大领导汇报，完成之后也有各种庆功宴等等可以一起吃饭。我说这些，并不是说多想跟领导套近乎，只是客观地说明你接触的范围会相当有限，所以交流思想和内容也会受限。跟 level 比你高的人交流，即使只是听君一席话，可能也收获颇丰，这是不争的事实。

还有一个侧面可以印证你的重要性，那就是看你能闯多大的祸。以一个互联网产品来说，你能偷到用户数据吗？你能删除数据库吗？你能让网站挂掉吗？你能让支付通道失败吗？显然，这对于 UI 工程师来说是不太可能的。我们可以让页面没有样式，但职业的素养要求我们在没有样式的时候也要保证可用；我们可以让页面的图标加载不出来，不过用户可能都不会发现... 所以，我们能闯的祸微不足道，那么我们对于产品所负的责任也几乎忽略不计，责任也就对应了我们的价值。

我们的角色定位，注定没办法直接体现到产品价值上。这也是焦虑的重要原因，工作没办法直接体现价值，就很难获得真正的成就感，你只有融入更大的团队通力合作产出的成果，才能稍微感受到一点成就。

UI 工程师 vs 前端工程师

前端工程师，应该算是这些年软件工程领域最成功的角色了。而 UI 工程师往往被认为是瘸腿的前端工程师。基本的认知就是：

UI 工程师 + JS 工程师 === 前端工程师

然而据我观察，现在趋势正朝着这样的方向发展：

JS 工程师 (write some CSS) === 前端工程师

当然，前端工程师到现在也没有十分明确的定义，我们做的都是大前端的子集，我们都称自己为前端工程师。那么问题来了，UI 工程师到底要不要往前端工程师上靠呢？到底算不算是个瘸腿的前端工程师呢？

既然前端工程师都没有明确定义，那么 UI 工程师当然也可以说自己是前端工程师，毕竟工作内容还是有交集的。但是作为 UI 工程师本身，以设计的思维和背景去对标工程师，真的有优势吗？就拿 CSS 这门“语言”来说，后台工程师根本不认为这是一门语言，只是一堆浏览器配置而已。同样 HTML 这种标记语言，也不过是一堆标记而已。

那么 SASS，LESS 这种预编译工具，是否应该在 CSS 设计之初就包含在语言特性中呢？我认为不是的。因为至少对我来讲，写 CSS 并不是编程的体验，而是在一张画布上慢慢勾勒作品的体验，我很享受那种让作品慢慢浮现出来的感觉。这也是 CSS 精准控制勾勒每个元素样式时才能体验到的。而 SASS，LESS 这种只是为了提高开发效率和降低维护成本，但却抹杀了 CSS 设计的本意。

我并不排斥这些工具，我也一直在用 SASS，毕竟日常工作还是效率为王。但我仍然无法想象，当修改一副“艺术作品”时，可以把相同的颜色一起改掉是什么体验。所以我认为，纯正的 UI 工程师是有着一颗设计师的心的，做艺术的发散思维与程序逻辑的严谨很难融合。同样，如果你热爱 UI 工程师这个岗位，也未必要往前端工程师上靠，因为你很可能会吃亏。

达摩克利斯之剑

我经历过公司级别的变卖与重组，深感在大趋势下普通员工的无奈与不知所措。那么，悬在 UI 工程师头上的达摩克利斯之剑到底是什么呢？那就是，低得可怕的天花板。为什么说低得可怕？因为从技术的角度讲，你的价值越来越难体现了。级别低的时候，可能还会搞些酷炫的动画创造些价值；级别高的时候，单纯 UI 层面产生的价值已经不足以匹配公司付给你的薪水了。

是因为 CSS 简单吗？当然不是，CSS 难得很，写好页面也没那么容易。但是，HTML 代码整洁、语义化、结构清晰，写的 CSS 严格还原设计稿，追求 1px 对齐，这些对于用户来说有意义吗？对公司来说有意义吗？许多用户甚至根本看不出 1px 的差别，你的这些追求可能仅仅是满足职业本身的匠心而已。那么，UI 工程师的价值到底在哪里呢？至少，在产品研发流水线上，我们是重要的一环，而且时间可控，很少出错。

那么天花板到底有多低？单从还原页面这一技能来说，我认为一年左右就可以无障碍还原任何页面了，两年左右就可以输出干净整洁符合标准，既照顾页面性能语义化，又能让为 JS 工程师考虑周到，让他们用得很舒服的静态稿了。两年之后呢？不怕，我们还有跨部门分享、行业会议、自研工具、性能优化等等，看上去能做的事情也挺多的，所以你心里踏实了吗？

自怨自艾的小媳妇

文章写到这里，我一直都是个自怨自艾的小媳妇的口吻，像是有一肚子苦水吐不完。其实这并不全是我想说的话，你可以把它当成第一人称的小说，我抛出一些问题和思考，我们共同探讨而已。从我本身的经历来讲，我在这个岗位上做得并不长，在我刚刚感觉

到天花板的时候，就幸运地逃离了这个岗位。所以我也很难体验到，坚守岗位并且突破天花板的那些同学是何感受。本文的目的肯定不是吐槽，而是发现和解决问题（如果真的存在的话）。所以，你可以选择继续读下去，也可以选择到此为止，因为后面的部分也是满满的负能量。

那些年的前辈们

在职业生涯的道路上，那些前辈们就是我们的指路人，从前辈的路线和发展上就可以大致看出我们未来的样子。设想你的岗位上比你大 10 年的前辈仍然没车没房，每天加班到深夜，那么可想而知，你在这个岗位上要付出多大的努力才能有出头之日了。

曾经我也有许多前辈，可是，他们都转行了。有的做了产品，有的做了交互设计，有的做了开发，有的去小公司当领导，有的创业当老板。总之，做什么都好，就是不要做网页重构了。

人们在面对自己看不懂的事情上，往往会本能地找个简单的理由安慰自己，比如他家里有关系啊，他本来就比较能忽悠啊什么的。但事实上，别人成功的原因往往都不只是我们看到的那些，只是我们不愿意承认而已。转了产品运营的，一定是平时对产品有更多的思考，对运营数据给予更多的关注，慢慢培养起来的感觉；去小公司当领导的，也绝不只是靠一张嘴忽悠来的，一定是对整个部门的流程体系把握得更清楚，即使没有太多实际经验，在人员管理及业务方向把握上也能说出个一二三来的；创业的就更不用说了，不知道背地里准备了多少个夜晚才能决心一搏。

我们做技术的，往往对技术之外的东西懒得去想懒得去关注，甚至会看不起那些技术一般，但却能靠拉动资源，笼络人心做出成绩的人。而现实生活中，往往是这些软技能比较强的人爬升得更快，一门心思钻研技术反倒成了束缚和瓶颈。

在任何人和人组成的社会关系里，沟通、交流、协作都是最重要的，而技术技能本身则没那么重要。技术好只能赢得同行的认可，但却未必都能带你走好职业生涯。越是技术好，越是深入钻研下去。而那些知道自己技术不行的，才会思考其他的出路，反倒成就了他们。当然，这样说可能有些偏激，毕竟很多成功的 CTO，架构师也很厉害，也算是功成名就了。只是，几乎还没有听说 UI 工程师出身的 CTO 或者 架构师。

向左走，向右走

聊到转行，UI 工程师有很多选择，既可以向左走（上游），也可以向右走（下游）。虽然说得轻松，谁都知道可以转行，可是说转就能转吗？这里，就要看你的个人准备，平台，以及外部资源了。

首先，你要对自己的现状有充分的认识，并朝着自己想要的方向准备着，比如你想转产品，就要在平时多关注一些产品细节，关注产品文档的撰写等基本技能，积累自己对各种互联网产品的观点和见解等。

其次，是平台。像腾讯阿里这种大平台，都有内部的转岗机制，只要另外一个部门愿意接收你，就可以直接转到其他岗位。即使不是真正的转岗，也有机会尝试轮岗一段时间，再回到自己的岗位，刚好可以考验一下自己是否真的热爱那个岗位，别再入错行哈哈。

最后，你的外部资源可以包含很多方面，类似达康书记的“政治资源”。人脉、影响力、口碑都是你的外部资源，在你的本职工作上，如果你还能表现出其他方面的能力并留下不错的口碑，或者你有关系过硬的同学等在其他公司的其他岗位身居要职，那就有可能帮你完成职业的转变。男怕入错行，女的也怕，如果你发现自己并不热爱这个岗位，那还是趁早另作打算。

全栈工程师 or 全栈设计师

UI 工程师的工作绝不仅仅是页面和 UI 方面的工作，也会把触手向上游或者下游延伸。向上游加上一些设计能力，可能就会成为全栈设计师。向下游多一些后端编码能力，可能就会成为所谓全栈工程师。“全栈”这个词其实是饱受争议的，我个人也不是太认可这种称谓。在核心的工作能力之外扩展一些其他的能力是很正常的现象，就像许多泥工瓦工也能做木工，甚至还能做水电，是一个道理，只是做得没有专职的那么精细而已。

那么互联网行业的各个工种也是一样，你可以说你都会，但是无论哪一样都一定比不上专职的人员，除非你的岗位要求就是每一样都会一点就够了。从这一点上讲，专职的 UI 工程师也是不可或缺的角色，当 UI 层面有什么疑难杂症，或者复杂的交互效果没人实现得了的时候，专职的 UI 工程师就体现了极大的价值。

全栈的价值在于，你在本职工作上已经是专家了，并且你还懂得其他角色的知识，即所谓的 T 型人才，这才是全栈的意义所在。你如果想做全栈，那么先找到一个方向深入下去做到专家级，在成为专家的过程中，也不忘补充其他相关的知识，这样慢慢你就可以成为一个全栈的人才。

而说到找方向，虽然说要遵从自己的内心，但是大部分人其实根本不知道自己想要什么，也不知道自己到底喜欢不喜欢这个方向，往往都是在摇摆不定中虚度了职业生涯早期的大好时光。对于前端开发来说，很多人不知道自己到底是喜欢重构，还是喜欢 JS 多一点，我只说一句，如果你真的非常想进 tx 的话，重构应该是一个相对（JS）较低一点的途径。先打进 tx 内部，扎扎实实做几年，再做打算。

必须转行才有出路吗

当然不是，我有许多前同事都坚守在自己的岗位，也都做到了管理层。问题是，这是不是你自己想要的。如果你就想认真做好 UI，直到从技术转管理，也是条不错的路线。这时，最重要的就是清楚游戏规则，或者叫玩法。

努力工作是最基本的，但是不是努力工作就一定有好的绩效。你要看到你的绩效重点在哪里，你的绩效又是如何影响你上级的绩效的。只有努力成就你的上级，保证他的绩

效，你才能有好的绩效。这里越说越像职场鸡汤了，再说有点投机取巧之嫌。总之，文章、分享、自研项目、工具、微创新、专利等等，能做的都拼命做就对了，这些都是实实在在的产出，一定是加分的。当然，要在做好本职工作的基础上，别连页面都没做好，连运营都搞不定，三天两头被投诉就完蛋了。

套路，全都是套路。核心思想就是，只靠搞关系是下策，让人瞧不起；有能力但不会表现，只顾埋头苦干的是中策；既有能力又会表现，甚至能为领导着想的，一定是发展最好的。

核心价值在哪里

如果我说 UI 工程师的价值比不上开发工程师，大家可能不服气。但是当年的页面重构（T 族）的薪水确实是比同期的开发工程师（T 族）低一个 level 的，具体低多少我不能说，或许现在已经持平了。好，我们忽略薪水的差异，来谈谈成就感。成就感是你感觉到自己的价值时才会有的，回想一下你最有成就感的项目是怎样的？是否是因为你的努力而起了决定性作用，才获得巨大成功的？如果答案是 yes，那么恭喜你，你真的很重要。

总之，在我的工作期间，我想不出多少在正常工作内容中让我很有成就感的事情。反倒是我们做了许多所谓的自研工具，来给产品和运营同学使用，反馈良好，在这里才找到了一些成就感。因为在这个方面，我们是自己产品的主人，我们决定着产品的形态、交互、使用细节等，好评是对我们极大的认可。相反，如果一个正常经由设计稿还原出来的页面，即使有再大的效果，可能也是设计师的杰作，而实际的页面只能无限逼近设计稿，但是却永远差那么一点儿。所以，还原得好是应该的，还原得不好，那就是水平不行了。

所以，作为一个 UI 工程师，我们的核心价值到底在哪里？快速产出页面？高保真还原设计稿？以技术的角度给出专业的交互实现建议？为设计、产品、运营开发便捷的工具？为页面增加酷炫的 CSS3 动画？解决各种疑难杂症？对了，说到点子上了。我也常常思考我自己的核心价值在哪里，要说专业能力，我曾经也是拿得出手的，但是这并不能成为我的核心竞争力。经过这两三年的洗礼，我想我唯一能自诩为竞争力的，就是我解决问题的能力。无论什么样的疑难杂症，烂摊子也好，只要我答应接手了，就一定能做出个令人满意的结果。

我想这样的解决问题的能力与我几年的页面重构工作不无关系，因为我身边还有很多优秀的同事，也表现出了类似的特质。因为我们是在困境中穷则思变的人，我们绞尽脑汁希望做出些不同的东西，借此提升我们微不足道的价值，这样的历练，不是每个角色都有机会体验到的。

所以我想说，UI 工程师应该是最容易培养出综合素质，培养出扎实的解决问题能力的一种人。我们有设计师般活跃开放的思维，有处女座般的专注细致，有程序员样的逻辑和严谨 ... 差不多不吹了 ... anyway，要对自己有信心，尽快找到自己的核心价值，并发扬光大。

叫什么不重要，是什么才重要

美工也好，UI 工程师也罢，别人叫我们什么，并不能让我们变成什么。幸好我们都喜欢自黑，也并不在意别人怎么说，切图也不丢人，想多了而已。我们三四年都是被构建构建这样叫过来的，也没觉得自己够贱啊，还顺便练就了强大的心理素质。归根结底，我们之所以怕别人叫我们美工，可能是源于内心隐隐的不自信。我们生怕自己真的就只是个美工，真的就只能切切图而已，所以这个词就成了抹不去的阴影、伤疤，一旦提起就像是在伤口上狂妄地撒盐，让我们不敢直视。本能的反应让我们想立马就回一句：你丫才美工！

当我们的内心真正强大起来的时候，就不会在意这些细节了。要知道，叫什么其实不重要，重要的是你觉得自己到底是什么，你自己有没有接受你的现状，有没有真正地接纳自己。所以，UI 工程师这个岗位到底好不好，还要你自己来决定。你热爱这份工作，那它就是好的；如果你不热爱，自己都不认可这份工作的价值，那它就是坏的。

破局，自我剖析

突破自我是最难的事情，那种被现实束缚的感觉就像是自己被包裹在一个超大的皮球当中，无论怎样拳打脚踢，它都能把你的力量消耗殆尽，让你继续包裹在里面不能动弹。很多人可能听说过 SWOT 分析法，其实也没什么特别的，我们都曾在某个对自己比较重要的节点分析过自己的优势劣势，机会和威胁。只是实际操作中，你未必能清楚自己的优势，也未必能承认自己的劣势，更未必有眼光能看到机会和威胁。那么这样的分析还有意义吗？当然，意义还是有的，经过这样一番自我审视之后，你还是能加深对自己的了解，以及对现状的把握。

如果分析之后，你能清晰地看到你可以把握的机会，那就朝着这个方向努力就没错了。如果你看不到机会，那也不要紧，可以先朝着你认为正确的方向去摸索，只要不是安于现状不肯努力即可。总有一天机会会慢慢浮现在你眼前。

我并不是忽悠所有 UI 工程师转行，客观的事实是，我的绝大部分前辈都转行了，也有继续坚守在原岗位，基本都做了团队 leader 的前同事。他们走过的路径，未必对我们有参考价值，每个人的情况都不同，没有什么可比性。但是值得我们学习的，是他们在华丽的转身之前，都做了哪些准备。可能需要你去观察，可能需要你去探访，如果你羡慕哪个转型成功的前辈，就想办法去听听他的建议。

没有什么真正的困境

在我们短暂的人生当中，并没有什么真正的困境，往往是一叶蔽目，想不清楚，患得患失。而当时过境迁之后，往往又觉得不过如此，这或许就是所谓的“历史局限性”吧。我们在特定的时间，特定的环境，就只能做出那样的选择。跳槽、转行也是一样，跳或者不跳，都是我们基于当下做出的最好选择，无所谓得失，因为人生没有如果，谁也不

知道如果之后会发生怎样的连锁反应。以我们有限的人生经验，我们只能相信“一切都会好的”，何况，UI 工程师的生活并不差，而且据我所知是非常注重生活品质的。

所以，既然没有什么真正的困境，那我到底在说什么呢？

是的，没有什么真正的困境，我们都在路上。我不是成功的例子，我的经历也没有什么借鉴意义。我抛出问题，说出我的思考，希望借此激发大家的一点思考。你可能不会在这里找到答案，但是一定有那么一两句话可以埋在你的心里生根发芽，最后长成参天大树。最后，借用一句万能甩锅金句：我说的都是错的。

GitChat