

总结与展望 | 低代码之路，我们才刚刚开始

2022-04-22 陈旭

《说透低代码》

课程介绍 >



讲述：陈旭

时长 09:42 大小 8.89M



你好，我是陈旭。

好快，这个专栏的常规更新部分到这里就正式结束了。恭喜你打败了学习的惰性，一路坚持到这里。现在，我们是时候停下来稍微总结一番，以便更好地开启后续的学习。

创建好了这篇文章的空白文档后，我的脑袋和新建的文档一样空白，思绪一下子飞跃到了四年前，我决定先和你分享下我是如何与低代码结缘的。听听我的故事，希望它能给你树立一些信心。

如果非要给我的低代码之旅设定一个明确的起点，那应该是 4 年前的 3 月，在我收到那封低代码相关的封闭研讨邮件的时候。但在这之前，我就建设低代码工具这个问题，与其他兄弟单位有过接触。受限于资源，双方决定采用分工合作的方式，我们负责 **Web** 组件，对方负责低代码引擎。

那时我们在成都组织了多次探讨，我现在许多关于低代码的设计理念，都是在这些探讨过程逐渐形成的。但可惜双方理念相差较大，最终没能走到一起。对方无法接受我先通用再具体的方案，执意要“务实”地对具体场景提供支持，因为这样“风险小、见效快”。

后来再收到这封闭关研讨的邮件时，我就明白老板这是下决心要自己搞了。

随着研讨越发地深入，我越觉得这个事情难度巨大，甚至，我还一度因畏难情绪和其他因素萌生了退意，但看着这个我当时花了近两年给老板画出来的饼，已经在我嘴边了，只要一张嘴就可以咬上一口，最终我还是选择了坚持。我至今仍在庆幸当时的选择。

从最开始设计 **Awade** 的编译器时，担心纯可视化搞不定，选择让编译器能够生成出人类可读、可改的代码；到后来慢慢走上正轨，解决了开发过程中大大小小的无数的难题；再到现在，脚下的低代码之路已经逐渐变得平坦且宽阔。从最开始受到的各种质疑，到现在，进度紧急时，经理们会特地点名必须使用 **Awade** 以确保其交付进度是可控的。

一时间，感慨万千。

最后的事实证明，先通用、再具体才是正路。通用性创造的是未来的可能性，虽然早期道路坎坷艰辛，但我们还是坚持下来了，路是越来越平坦的。反之，如果你过于着眼于眼下，看似务实，实际上更容易因架构设计的韧性不足，导致路越走越窄，举步维艰。我一直非常注重竞品调研，我搜遍公司内外各种竞品，做了很多调研，后来成都那个团队的产品也没有再出现在我的竞品清单中。

在这个专栏的交流群中，我看到很多同学都是一线低代码平台的设计者和实现者。可能很多和那时候的我一样，或是正绞尽脑汁给领导画饼，或是正迷茫在架构策略的抉择中，又或者是正受困于某个技术难题，不知如何向前。

我非常希望这门专栏能帮助你解决眼下的困境，也非常希望你能坚持一下，再坚持一下，走到最后。如果你在此过程中有啥困惑或难题，欢迎来找我。

课程设计的背面

时间拉回今年春节前，极客时间团队的小伙伴开始和我接触，邀请我来写低代码这门专栏。我当时的第一想法是非常兴奋，不谦虚地说，我是国内最早开始“吃”低代码这只“螃蟹”的一批

人，坚持到现在，已经超过 4 年了。在此过程中我积累了大量的实战经验，极客时间是整理和展示我这些经验的一个很好的平台。

兴奋之余，我又感觉非常为难：具体的内容应该如何编排呢？毕竟，低代码技术是一个综合性非常高的话题，低代码涉及的方面非常多，可以讲的内容也非常多。我们可以从实际应用的角度来讲，也可以从系统架构实现的方向来讲。同时不同的方向上还可以有不同的侧重，不同的方向和侧重，面对的人群还不一样，适用的内容也不一样。

所以，我花了非常多的时间纠结到底怎么挑选内容。**最终，我选定了低代码平台的架构和实现这个大的讲解方向，同时偏重于系统架构的设计，忽略掉过细的实现细节。**

那么，在低代码的架构和实现这个大方向下，要选择哪些功能和模块进行讲解呢？

经过一番精挑细选后，我决定围绕着低代码编辑器这个功能来打造这门课。低代码编辑器是低代码平台最关键的功能。它的能力和实现的质量，直接决定了低代码平台的能力和质量。虽然低代码平台综合性很高，可以做的功能点众多，在各家企业里落地时，大家都各有侧重，有的先解决服务端侧，有的先解决前端侧的问题。但无论你侧重建设哪个端的功能，低代码编辑器都是绕不过去的一个功能点。

这门专栏的常规更新部分的主体内容，就是按照这样的考虑来设计和编写的。

但是，我们当然不可能一上来就单刀直入，直接剖析低代码编辑器，还是要遵循学习知识的一般认知流程。所以我将内容分成了三大部分，先从认知基础与架构策略切入，着重介绍了低代码的演进策略和低代码编译器与编辑器之间的关系。

我希望你在开工之前先想好路怎么走，这比啥都重要。你要先想清楚你负责的子系统与上下游子系统的关系，若保持强耦合，对方值得托付吗？还是留个活扣，给自己以后多一个选择？特别是，你不要急着编码，一旦开始写代码，就容易陷入细节，不能自拔了。**写代码永远是最容易的一件事，前提是你真的想清楚了。**

接下来的第二、第三部分是我们常规更新的重点内容：低代码编辑器的架构和实现，以及低代码平台的拓展。这些内容都比较硬核，因为我们很难用短短几千字讲清楚一个架构知识点，而且抛开业务谈架构都是在耍流氓，你可以看到，每一讲中我都非常注重业务场景的讲解，也针对不同的业务场景给出了一些建议。你可以结合自己的实际情况辩证地看。

我注意到，前两天交流群里有同学提到，每一讲的文字量都很大。其实，我之前写文字稿时就注意到了这个问题，我往往需要用目标篇幅的 **1.5** 倍左右才能完成一讲的内容，常常才写完第一个小标题时就发现已经用掉了大半的目标篇幅了，不得不回头再精简内容，但总是觉得，少了这块不行，少了那块也不行。

即使我“精打细算”地使用好每一讲的篇幅，但你应该也可以发现，有些内容依然是蜻蜓点水般一带而过了，没能讲透，特别是第三部分的两讲更是如此。这些没涉及、没展开、没讲透的内容，就要留着在这门专栏的动态更新部分才能展开了。

动态更新是极客时间的一种创新，特别适合低代码这样的综合性高，内容繁复多样的知识的学习。和常规更新阶段事先敲定内容的学习形式不同，动态更新阶段中，你可以一边学习，一边和我互动，我会针对你的学习状况和需求，动态设计和组织剩下的内容，达到定制化的学习效果。

动态更新内容，将由你来定制

那么动态更新部分，将会有哪些内容呢？

- 首先，这一阶段最主要的是**把在常规更新部分里的那些“等有机会...”“下次再说...”的部分补齐**。你可以将这个课程看成是一棵树。常规更新部分是大树的树干和部分枝丫，而动态更新部分是沿着树干生长出的各个重要枝丫；
- 其次，这一阶段我们要尝试跳出以低代码编辑器为中心的思路，**尝试围绕其他角度，来完成低代码平台的架构和实现**。因为低代码编辑器并不是低代码平台的全部，它只是一个核心部件，低代码平台还有其他许多有价值的功能需要建设；
- 最后，我们还会**对业界进行持续跟踪和观察**，包括但不限于开源社区、调查机构、甚至还有一些竞争对手的资料，但这个角度的内容篇幅不会很大。

动态更新内容的大方向，将和常规更新的内容保持基本一致，我将继续保持在低代码平台的架构和实现这个方向上进行讲解，不会做方向性的大幅度调整。

不过，此时此刻，动态更新的内容还没完全定型，你我可以一起来设计剩余的内容。欢迎你把你想要学习的内容写在评论区、[🗨️ 交流群](#)，或者这个 [📝 调查问卷](#) 中，我们一起来设计剩余的内容。

正如在开篇词中向你承诺的那样，动态更新部分将有 20 讲左右的内容，大致以每个季度至少一讲的频率，每年更新 5 讲，持续 4 年。衷心邀请你和我一起完成这场“长跑”，在接下来的学习中，不仅紧追低代码的前沿“脉动”，更重要的是在逐渐深入的学习和实践中，完善低代码的知识体系，提升自己的架构能力。

这里先预告一下，接下来动态更新阶段的第一篇文章将在**今年 7 月**更新，一定要记得回来学习呀，我也会在 [交流群](#) 里通知你的！

写在最后

最后，我还想跟你说个题外话。有可能你曾经去词典里搜索过“awade”这个单词，想必是一无所获。因为这是一个我自创的单词，它来自“Anyone can be A Web Application Developer Exper”这句话的首字母，翻译过来就是人人都是 Web 应用开发专家。


这就是 Awade 的使命，也是当时在研讨和设计 Awade 过程中，大家共同的愿望。作为一个低代码平台的设计和实现者，我们希望在它的帮助和赋能下，人人都可以成为一个 Web 应用的开发专家，而不是仅仅只有那些掌握了 Web 研发技术的职业开发人员才能成为专家。


现在，我把 Awade 的架构和实现方法整理出来了，希望你通过对这门专栏的学习，也能设计出一个低门槛、高效率的低代码开发平台，赋能它的使用者，让大家都成为 Web 应用的开发专家，为业务实现真正的降本增效。

探索低代码之路，我们才刚刚开始，就让我们一起继续研究，探索出更好的低代码实现之路吧！

分享给需要的人，Ta 订阅超级会员，你最高得 50 元

Ta 单独购买本课程，你将得 20 元

 生成海报并分享

 赞 0  提建议

精选留言

写留言

由作者筛选后的优质留言将会公开显示，欢迎踊跃留言。