

项目综合管理：从一个失败的项目说起需求和设计 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/260/2021_2022__E9_A1_B9_E7_9B_AE_E7_BB_BC_E5_c41_260052.htm 由于是一个小项目，感觉需求也简单，再加上时间紧，如果从需求开始一步步来，时间肯定来不及，在这种情况下，项目就匆匆的开始了。为了节省时间，分层、设计等等都不去考虑了，想到哪写到哪，完全瀑布式开发。直接结果是，完工时间一拖再拖，最后不得不决定下一版本整个推倒重来。项目失败的原因有两个：需求分析不到位、架构设计不合理。需求和架构设计是相辅相成的，如果需求分析做的好，架构设计合理，那么就可以灵活的适应变化的需求，这是理想的状况。如果需求好了，架构有不合理的地方，项目也可以实现，只是以后的维护会有困难。架构好了，需求没有做好，随着需求的进一步完善，项目也会完成，这是容易实现的一种状况。如果都没有做好，象这个项目一样，就只能有两种选择：1、尽早重来，这样虽然会推出项目的完成日期，但总的来说还是省了时间，也可以交出一个满意的项目。2、通过架构的修补、需求的完善，先在规定时间内交出一个不完善的版本，等下一个版本在重新开始。好的需求，会加快项目的进度，也可以给开发人员的设计提供帮助。项目开始前一定要做好需求和设计，至少要有明确的思路，匆忙开始的项目很可能会失败，至少也会走弯路，而走弯路花的时间很可能会超过在需求和设计上省下来的时间，更不用说失败的项目所造成的后果。需求和设计难做时，也可以先动手实现一个版本，让客户体验，这样可以防止设计上迷失方向，在设计上走错路。

小型项目是不是还需要设计？我以前一直认为，项目小，完全可以不做任何设计，类、接口等都不用去想，分层也是不必要的，因为做设计会花时间，实现设计也会花比直接写代码（如双击按钮，在事件中写代码，而不去掉用已经封装好的实现。）花更多的时间。简单有效的方法就是边想边写，这样可以有最快的开发速度。但通过这个项目，我认识到，小的项目，也可能存在你现在还没有发现的陷阱，如果采用上面的开发方式，遇到陷阱时就不能灵活应对，因为架构不好。更不用说，经常要变化的需求，等需求一变，又是头痛的时候，还是因为架构不好，最后设计越来越糟，到处都是重复的代码，时间在逐渐的流逝，而项目的进度会越来越慢，你最后发现，要想让项目完美的实现已经不可能了，时间都浪费在一些重复的劳动上，而有好的架构，这些都是可以避免的。人也会变的绝望，面对着象线团一样的代码，根本没有了工作的激情，也没有了动力。再加上一些小项目有时候会牵扯到几个系统，按照这种设计，根本没有办法进行单元测试，集成测试又会耗费太多的时间，有时根本就不能测试。结果只有痛苦的重新开赛。而有好的设计，情况就会完全不同，开始会慢一些，而随着项目的进行，一切会逐渐的明晰，你也不会惧怕陷阱和需求的变化。看着项目在一天天的完善，心情也会开心。最好项目会成功。因此，即使是小的项目设计也是十分必要的。什么是好需求？需求要从客户的角度去寻找，需求是客户要求的抽象，而不是具体的表现，这样做的需求才能对以后的设计产生积极的影响。而一些具体的要求可能都是易变的，这些可能是商业政策，而不是真正的需求。需求总是易变的，这就要求架构要有灵活性，

灵活性不是靠提前设计实现“你认为将来会有的需求”，而是靠抽象，这样可以在需求变化时，架构做最少的修改。从开发者角度说，需求是架构必须要实现的要求，要把抽象的需求再扩展到具体。这样需求就经历了从具体（客户的描绘）到抽象（架构，好的需求）再到具体（实现）的一个过程都是自己的理解，有不合理的地方请指教，刚才图书馆借了一本关于需求的书，以前对需求了解太少了，上学时学的印象不深，只有碰到问题才能加深理解。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com