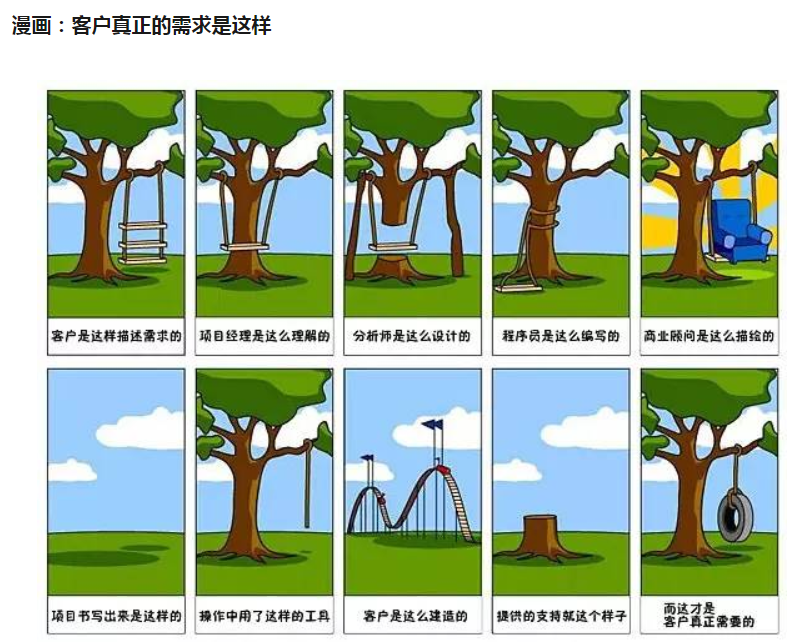
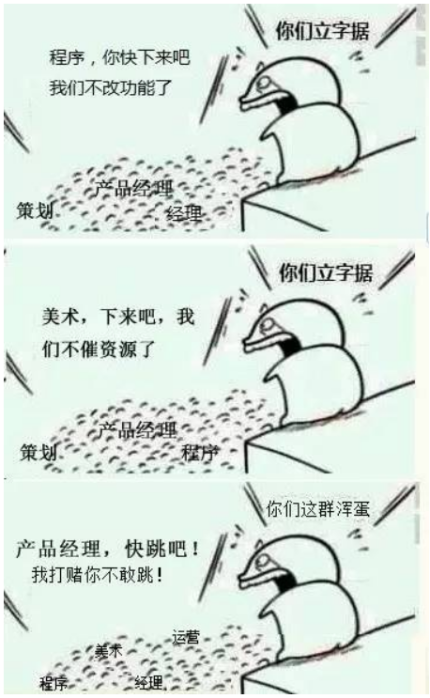
# 关于“杀死一个程序员只需修改三次需求”的想法，分析和建议

这句话出自于“关于程序员的经典段子”，类似的还有：再好的香水，干不过韭菜盒子。再好的网名，干不过备注。再好的代码，干不过需求变更！，更有许多漫画来调侃程序员和产品之间不可调和的矛盾，例如：



通过这些看似调侃又透露出深深无奈的段子，我们可以看出，如何正确准确的获取需求，减少后期改动概率，成了程序员最重要的问题。

首先，我们需要弄清需求分析是什么：软件[需求分析](https://baike.baidu.com/item/%E9%9C%80%E6%B1%82%E5%88%86%E6%9E%90" \t "https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E9%9C%80%E6%B1%82%E5%88%86%E6%9E%90/_blank)就是把软件计划期间建立的[软件可行性分析](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E5%8F%AF%E8%A1%8C%E6%80%A7%E5%88%86%E6%9E%90/10986831" \t "https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E9%9C%80%E6%B1%82%E5%88%86%E6%9E%90/_blank)求精和细化，分析各种可能的解法，并且分配给各个软件元素。需求分析是软件定义阶段中的最后一步，是确定系统必须完成哪些工作，也就是对目标系统提出完整、准确、清晰、具体的要求。开发软件系统最为困难的部分就是要准确说明开发什么。最为困难的概念性工作便是要编写出详细的技术需求，这包括所有面向用户、面向机器和其它软件系统的接口。如果做错，这将是会最终给系统带来极大损害的一部分，并且以后再对它进行修改也极为困难。

所以，面对如此重要的工作，其中的每个环节我们都要细致准确的做好，进行需求分析时，应注意一切信息与需求都是站在用户的角度上。尽量避免分析员的主观想象，并尽量将分析进度提交给用户。在不进行直接指导的前提下，让用户进行检查与评价。从而达到需求分析的准确性。对于用户的需求要进行鉴别、综合和建模，清除用户需求的模糊性、歧义性和不一致性；要将对原始问题的理解与软件开发经验结合起来，以便发现那些要求是由于用户的片面性或短期性行为导致的不合理的要求，找出真正有价值的潜在需求；在评审阶段，要在用户和软件设计师的配合下对需求规格说明和初步用户手册复核，确保软件需求的完整、准确、清晰、具体，并使三方的理解达成一致。一旦发现遗漏点或模糊点，尽快更正，再行检查。

只有做到这样的充分沟通，严谨客观的逻辑思考，反复核查，才能有效减少程序员自杀的概率。