**软件体系结构 第三次作业**

**参考答案**

**1.阅读软件架构师应该知道的97件事，选择3件印象最深的并写出感想。**

答：

**1． 不要在一棵树上吊死**

为客户提供多样化的解决方案。没有一种方案能解决所有的问题。因此软件架构师应当提出多种方案，并为客户提供各种方案的优缺点。在一棵树上吊死，在软件架构中意味着只执着于一种方案，在这个方案行不通或者不够好的不去采用新的方法去解决问题，这样只会把问题越弄越大，情况越变越糟，软件开发的进度也随之停滞甚至倒退。所以，灵活变通，多样化的解决方案，不仅是良好沟通的开端，更给软件开发打下坚实基础。

**2． 向建筑师学习**

借鉴建筑行业的经验。架构师应该蕴含适当的艺术成分！亨利·沃顿说，盖房子和别的手艺一样，所有努力都是为了结果，为了完成令人满意的工作。合格的建筑应该符合以下三个条件：实用、坚固、令人愉悦。约翰·拉斯金说，建筑师首先应该是伟大的雕塑家，或者伟大的画家，否则他不过是个建筑式工人。众所周知，在软件工程当中的架构师和建筑工程当中建筑师有许多相通之处，都是负责“产品”的架构设计。作为一名合格的架构师，在规划软件架构的时候不应当只是搭一个简单的框架满足功能需求就结束，因为这是传统的码农而不是架构师。合格的架构师应当从多个不同抽象层面理解软件系统，将其艺术地建构起来，从而形成简洁明了又具有实用性的软件架构。

**3． 命名要恰如其分**

弄清楚要做的究竟是什么。设计就是要去实现各种意图，例如，快速、廉价、灵活、而名字便是用来承载和传达这些意图的。正常情况下，一个好的程序员应当遵守的规定的命名守则。变量名应该能够描述变量表示的信息；优先考虑代码的可读性和质量而非数量；采用标准的命名约定，做出一个全局决策而不是多个本地决策。在我个人的实践中，程序并非像一次性用品那样使用结束就直接丢弃。如果出现需要后期维护或debug，在命名上的偷懒将会带来更大的麻烦，需要花很大的力气来理解自己当时写的代码，使得效率低下，甚至需要推倒重来。因为，良好的命名习惯非常重要！

**4. 对最终用户而言，界面就是系统**

软件产品的使用者大多数情况下并不是专业的开发人员，他们对于软件的第一印象就是软件界面。如果在软件开发过程中忽视用户交互，造成界面的糟糕呈现和不够良好的用户体验，就会使得软件产品的使用者认为这个软件产品不够“优秀”。事实上，在现实生活中我们也经常会因为某个应用的界面设计不佳或者使用不够“顺手”，而放弃它。因此在软件开发中也应该注意要为客户提供好的用户界面。

**5. 性能至上**

提高交互性能才更符合用户对于优秀软件产品的期望，试想一个应用如果在点击功能后需要长时间的等待，这个应用迟早也会被用户抛弃。每年的双11，淘宝软件的抗压性能也是双11活动能持续多年的一个原因，倘若淘宝性能不佳在一次双11活动中就崩溃无法继续，使得用户的双11购物一次次的失败，广大用户也就会在一次次的失败中放弃双11。因此在软件开发中我们需要分析必要的交互步骤，可以挖掘更人性化的设计思路，提高交互性能。

**6. 没有永不过时的解决方案**

因为硬件，技术，软件，都在不断的变化。有新的硬件，新的平台，新的语言技术的诞生，所以解决方案只能不断的更新，更新到极致再脱胎换骨，换一个解决方案。只有如此才能更好地解决实际问题。

**7. 客户的客户才是你的客户**

假设你正在编写一个电子商务应用程序，那么该仔细考虑的应当是那些会在那个网站上购物的人的需求。客户的客户才是我们开发软件的最常使用者，这给我们的需求分析带来了非常重要的启示。尽管给软件开发者付费的是客户，但客户本身也极有可能会对他们客户的需求有所遗漏。我们不能一味按照客户的要求去做，应该遵循实践所得的规律，为产品综合考虑，与客户沟通达成一致，争取各方面都满意的结果。只有真正将最终使用者的需求落实，才能实现客户与“客户的客户”的双赢，从而达到开发者与客户的双赢。