

敏捷首先要“看得见” PDF转换可能丢失图片或格式，建议  
阅读原文

[https://www.100test.com/kao\\_ti2020/277/2021\\_2022\\_\\_E6\\_95\\_8F\\_E6\\_8D\\_B7\\_E9\\_A6\\_96\\_E5\\_c41\\_277570.htm](https://www.100test.com/kao_ti2020/277/2021_2022__E6_95_8F_E6_8D_B7_E9_A6_96_E5_c41_277570.htm)

在项目之初，曾想到在项目里面导入敏捷开发的一些好的实践，但是仔细分析了一下，项目组的成员大多在经验上还不是很丰富，硬要上敏捷的话，会得不偿失。考虑来考虑去，还是一点点来。先从“看得见”做起。所谓“看得见”就是让任何需要管理的东西，要让大家都看得见，类似于丰田的看板管理，反过来说，任何看不见的东西都是无法管理的。这个现在在日本是一个很流行的管理理念。特点就是简单实用。我们的实践如下：

1、让每天的工作内容看得见。每天早上来，大家围在白板前开一个短会。根据日程表把各自今天要完成的内容写在白板上，按人名分类，按优先级排序。每一项任务前面标志一个叉，表示还没完成。会开后使用白板自带的打印机打印出来，用磁石粘在一个大家都能看到的位置。这样做的好处就是，各自的每天的任务明确，信息共享，每个人今天需要看什么谁都知道，同时也有利于Leader的管理。

2、频繁沟通。对照白板的内容，对每个人的工作我会每天询问两次。中午一次，下午4点一次。中午那次主要询问有没有什么问题，对于完成这件工作的一些想法的交流，下午4点主要就是询问进度了。这两次的目的是不同的，中午的那次，主要是通过想法交流，我们一起来保证我们做的是正确的事情同时采用了正确的方法。下午那次，是为了调配各自的任务量，让工作已经完成的组员去帮助工作还没有完成或者有问题的组员。有的人可能会想，别人的工作为什么让我去做。在项

目之初我就强调了，我们是一个团队，一个人的工作没完成，等于我们大家的工作都没完成，项目无论什么地方出了问题，都是我们大家的问题。还好，我的组员在这一点上执行的非常好。

3、设计程序先想测试。测试先行，估计好多人都在实践了，不过说实话，我个人觉得实践起来不是那么简单。所以就采取了一个折衷的做法。我要求组员设计程序之前，不需要写测试程序，但是一定要先想如何测试。大家一起讨论或者Review一种技术方案的时候，负责审核的一方都会强调，先考虑能不能测试，怎么测试。我的想法是无法测试的程序也就是没有意义的，因为谁也不知道它正确与否。知道怎么测试了，程序写起来就快多了。这是我们的体会。

4、制作工具不偷懒。项目的前期，我们就考虑到需要一种模仿用户下订单的工具。在研究完业务之后我们就做了这么一个工具，花费了一些时间，但是事实证明，这个工具非常的有用，无论是后来的功能上的测试还是负载测试，这个工具都起到了很大的作用。所以，在项目前期，工作不是很忙得时候，做一些以后会用到的工具是非常好的一个实践。关键就看能不能不偷懒了。

5、定期总结。定期我们会总结一些前期的工作，采取Keep, Try, Problem分类的方式。同样是写在白板上。Keep的内容是需要继续坚持的，Try是对一些问题的解决方案，需要下一阶段实行并验证其效果的，Problem是存在的问题，需要改进。定期总结的作用就是强行将我们从程序堆里面拽出来，重新审视我们的工作。我对组员的要求是不仅要会写程序，更要会工作。敏捷不是理论，是各种各样的实践。最重要的是，这些实践都适用于自己。

100Test 下载频道开通，各类考试题目直接下载。详细请访问

