

3.软件管理沉思录读后感

3.1.软件质量

在软件开发过程中, 为了保证质量始终如一, 必须遵循以下 8 个步骤:

1. 确立质量控制的策略、目标和计划;
2. 正确训练、指导和支持开发人员及其团队;
3. 确立和维护软件需求的质量管理过程;
4. 确立和维护软件工程过程的统计控制;
5. 审查、检查并评估所有的产品制品;
6. 评估所有缺陷, 加以更正并用以识别、纠正和预防其他类似问题;
7. 确立和维护配置管理和变更控制系统;
8. 持续改进开发过程。

提到软件质量, 我们总是不由的想到软件测试. 然而, 实际上软件测试只是软件质量保证的一个手段, 并且这种手段在越来越复杂的代码规模下, 显得越来越力不从心. 之前在公司的实习体验告诉我, 软件质量从管理的角度更容易维护, 比如上述第五点: 我们常常发现CodeReview给软件质量的提高并不仅仅是针对代码的质量, 我们通过CodeReview还能发现很多bug, 其中很多bug很难通过测试测出, 也有很多目前不是bug但它对未来可能造成质量风险.

实践证明, 上述方法还可以帮助我们解决时间和金钱. 上述方法, 不仅仅是对测试人员的要求, 也是对全体成员的要求. 它让软件质量存乎每个人每个流程之中, 让全体人员遵循规则提高质量, 软件质量不再是测试与项目主管几个人的事情, 进一步降低了团队沟通成本, 提高了团队的一致性.

3.2.团队管理

没有一些共有的目标, 人们就不会努力他们只是投入时间。

书中有一段关于凝胶型团队的介绍:凝胶型团队是指紧密联系在一起的一个群体, 其密切程度使他们作为一个整体时迸发的力量超过了其组成部分之和。这种团队的效率要比人员构成相同、但没有形成凝胶型团队的群体大得多。同样重要的是, 凝胶型团队中的成员从他们工作中获得的快乐, 要比一般想象中工作本身赋子的快乐更多。

无疑,凝胶型团队是我们每个人梦想的团队, 然而在现实中常常有些问题会让团队变成一堆散沙, 这些问题可能涉及领导、合作、参与、拖延、质量、功能蔓延和评估等方面。

在我经历的项目中, 最严重的就是合作,拖延,质量,功能蔓延带来的问题.这些问题会浪费团队的时间并且削弱它们的活力,降低团队的士气. 在公司实习的时候, 有一条产品线刚刚成立, 很多老员工加入, 他们疯狂的迭代, 然而由于产品线之间资源不容易共享, 产品经历需求变动大, 疯狂迭代加班导致代

码质量低下, 团队迭代常常拖延. 最终这个产品线被合入其他产品线, 大部分工程师被也失去了原有的活力, 在领完年终奖后选择跳槽走人(发完年终奖后那个产品线的人基本全跳槽了...).

一支团队要想成为高效团队, 所需要的不仅仅只是恰当的任务和工作环境, 它的任务必须是重要的, 所处的工作环境必须有助于团队协作。团队必须直面一个艰巨的挑战, 而且要有空间去计划和管理其任务。通过为团队提供以下四项额外的支持, 上述需求就会被满足, 这就是: 凝聚力、目标、反馈、共同的工作架构。

1. 凝聚力:配合默契、彼此尊重、相互支持
2. 目标:目标必须是明确的和可量化的
3. 反馈:了解他们自己的表现, 能看到他们向目标前进的进度
4. 共同的工作架构:每个团队成员理解他们的角色和责任

那么我们如何让团队满足上述要求, 从而成为一支凝胶型团队呢? 首先, 团队交流至关重要, 这里的团队交流不止是普通意义上的交流, 它还指团队在各个方面的一致性. 比如我们可以通过一整套CI/CD流程, 保证开发运维测试人员的一致性, 让他们之间有共同的术语/目标, 从而提高他们的一致性, 进一步降低沟通成本. 而在传统交流方面, 要保证透明, 倾听和协商的原则: 透明保证了所有成员对具体情况都了如指掌; 倾听和协商则保证成员能得到反馈与相互尊重.

3.3.管理你的领导

“你来负责这个项目, 我希望3个月后得到一个结果.”这时候一定不要沉浸在负责这个项目的喜悦中. 如果你没有提出意见的话, 他们就会认为你能再3个月能完成. 这种行为是一种极不负责的行为, 你必须首先让管理者相信你项目会花费多长时间, 其次你应该制定一份计划, 并且是一份非常详细的计划, 有了这份计划你就可以确定真正的交付时间, 然后你需要和另外展开讨论, 介绍你的计划并让他们相信你的计划.

书中有一个很有意思的部分: 当一个项目注定失败时要做什么, 作者给出三个选择:

1. 继续埋头工作希望情况会得到改善
2. 寻找另一份工作
3. 试图解决问题

继续埋头工作可能很重要, 但是它并不会让情况改善, 并且这种态度是惰性的一种体现, 惰性是放弃的一种表现.

寻找另一份工作, 可能让你的简历变得糟糕, 毕竟你的简历看上去是由一系列失利组成的.

试图解决问题, 但注意这时你必须非常小心. 可以假设管理者处于这三种情况之中.

1. 管理者已经怀疑项目有了问题和麻烦。管理者有很多渠道来获取信息, 同时管理者通常具备一种本能的直觉, 甚至在别人告诉他们之前就已经嗅到了麻烦的存在。如果管理者曾经参与过几个项目, 他们就会知道有麻烦的项目会呈现出明显的特征。人们开始看上去焦躁不安、忧心忡忡, 笑声和玩笑消失了, 进展报告变得模糊。
2. 管理者想得到的是解决办法, 而不是问题。管理者很忙, 如果你带给管理者的是又一桩问题, 可以想象, 他们会避之唯恐不及。相反, 如果你表示可以提供帮助, 那么迎接你的可能就是热情的双臂。

3. 管理者不想有竞争对手。你的管理者负责你的工作分配、评估、报酬和晋升。如果他认为你在支持他，他就会帮助你解决项目中遇到的问题。不过，如果怀疑你是他的竞争对手，或者认为你已经进入了高层管理者的视线，不难想象，你会被半途拦下。

如果让我选择, 我会选择试图解决问题如果失败后寻找另一份工作(或换到另一个部门)