

08 | 以终为始，如何才能做好测试计划？

2018-07-16 茹炳晟

软件测试52讲

[进入课程 >](#)



讲述：茹炳晟

时长 15:22 大小 7.04M



在上一篇文章中，我为你介绍了如何高效填写软件缺陷报告，并为你解读了缺陷报告中的关键内容。今天，我将为你介绍一份成功的测试计划应该包含哪些内容，以及如何才能做好测试计划。

软件项目，通常都会有详细的项目计划。软件测试作为整个项目中的重要一环，也要执行详细的测试计划。正所谓运筹帷幄之中，决胜千里之外，强调的就是预先计划的重要性和必要性。

在早期的软件工程实践中，软件测试计划的制定通常是在需求分析以及测试需求分析完成后开始，并且是整个软件研发生命周期中的重要环节。

但是，在敏捷开发模式下，你可能会有这样的疑问，软件测试计划还有那么重要吗？我所在的软件项目压根儿就没有正式的测试计划，不也没出什么大问题吗？

的确，对于很多非产品型的互联网公司，由于采用了敏捷开发模式，的确很少去制定传统意义上的测试计划了，但这并不是说它们就不再制定测试计划了。

只不过是，测试计划的表现形式已经不再是传统意义上庞大的、正式的测试计划文档了，而更多的是体现在每个迭代（sprint）的计划环节，而且这样的短期测试计划可以非常迅速地根据项目情况实时调整。

所以说，**测试计划依旧存在，只是从原来的一次性集中制定测试计划，变成了以迭代的方式持续制定测试计划。**但是对于传统软件企业，或者是做非互联网软件产品的企业，它们通常还是会有非常正式的软件测试计划。

由此可见，无论对于早期最具典型性的瀑布开发模型，还是现在的敏捷开发模型，测试计划的重要性始终没有发生变化。那么，你可能会问，测试计划的重要性到底体现在哪些方面呢？在回答这个问题之前，我先跟你聊聊如果没有测试计划会带来什么问题。

没有测试计划会怎么样？

如果没有测试计划，会带来哪些问题呢？

1. 很难确切地知道具体的测试范围，以及应该采取的具体测试策略；
2. 很难预估具体的工作量和所需要的测试工程师数量，同时还会造成各个测试工程师的分工不明确，引发某些测试工作被重复执行而有些测试则被遗漏的问题；
3. 测试的整体进度完全不可控，甚至很难确切知道目前测试的完成情况，对于测试完成时间就更难预估准确的时间节点了；
4. 整个项目对潜在风险的抵抗能力很弱，很难应对需求的变更以及其他突发事件。

从这些问题中，你可以逆向思维推导出，一份好的测试计划要包括：测试范围、测试策略、测试资源、测试进度和测试风险预估，这五大方面，并且每一部分都要给出应对可能出现问题的解决办法。

测试范围

顾名思义，测试范围描述的是被测对象以及主要的测试内容。

比如，对于用户登录模块，功能测试既需要测试浏览器端又需要测试移动端，同时还考虑登录的安全和并发性能相关的非功能性需求的测试等。

测试范围的确定通常是在测试需求分析完成后进行，所以确定测试范围的过程在一定程度上也是对测试需求分析的进一步检验，这将有助于在早期阶段就发现潜在的测试遗漏。

同时，由于不可能进行穷尽测试，而且测试的时间和资源都是有限的，所以必须有所取舍，进行有针对性的测试。**因此，测试范围中需要明确“测什么”和“不测什么”。**

测试策略的话题

测试策略简单来讲就是需要明确“先测什么后测什么”和“如何来测”这两个问题。

病有轻重缓急，测试也是一样的道理，重要的项先测，而不重要的项要后测。**测试策略会要求我们明确测试的重点，以及各项测试的先后顺序。**

比如，对用户登录模块来讲，“用户无法正常登录”和“用户无法重置密码”这两个潜在问题，对业务的影响孰轻孰重一目了然，所以，你应该按照优先级来先测“用户正常登录”，再测“用户重置密码”。

测试策略还需要说明，采用什么样的测试类型和测试方法。这里需要注意的是，不仅要给出为什么要选用这个测试类型，还要详细说明具体的实施方法。

第一，功能测试

对于功能测试，你应该根据测试需求分析的思维导图来设计测试用例。

主线业务的功能测试由于经常需要执行回归测试，所以你需要考虑实施自动化测试，并且根据项目技术栈和测试团队成员的习惯与能力来选择合适的自动化测试框架。

这里需要注意的是，你通常应该先实现主干业务流程的测试自动化。

实际操作时，你通常需要先列出主要的功能测试点，并决定哪些测试点适合采用自动化测试，并且决定具体使用什么样的框架和技术。

对于需要手工测试的测试点，你要决定采用什么类型的测试用例设计方法，以及如何准备相关的测试数据。

另外，你还要评估被测软件的可测试性，如果有可测试性的问题，需要提前考虑切实可行的变通方案，甚至要求开发人员提供可测试性的接口。

第二，兼容性测试

对于兼容性测试来说，Web 测试需要确定覆盖的浏览器类型和版本，移动设备测试需要确定覆盖的设备类型和具体 iOS/Android 的版本等。

你可能会问，我要怎么确定需要覆盖的移动设备类型以及 iOS/Android 的版本列表呢？这个问题其实并不难：

如果是既有产品，你可以通过大数据技术分析产品的历史数据得出 Top 30% 的移动设备以及 iOS/Android 的版本列表，那么兼容性测试只需覆盖这部分即可。

如果是一个全新的产品，你可以通过 TalkingData 这样的网站来查看目前主流的移动设备，分辨率大小、iOS/Android 版本等信息来确定测试范围。

兼容性测试的实施，往往是在功能测试的后期，也就是说需要等功能基本都稳定了，才会开始兼容性测试。

当然也有特例，比如，对于前端引入了新的前端框架或者组件库，往往就会先在前期做兼容性评估，以确保不会引入后期无法解决的兼容性问题。

兼容性测试用例的选取，往往来自于已经实现的自动化测试用例。道理很简单，因为兼容性测试往往要覆盖最常用的业务场景，而这些最常用的业务场景通常也是首批实现自动化测试的目标。

所以，我们的 GUI 自动化框架，就需要能够支持同一套测试脚本在不做修改的前提下，运行于不同的浏览器。

第三，性能测试

对于性能测试，需要在明确了性能需求（并发用户数、响应时间、事务吞吐量等）的前提下，结合被测系统的特点，设计性能测试场景并确定性能测试框架。

比如，是直接在 API 级别发起压力测试，还是必须模拟终端用户行为进行基于协议的压力测试。再比如，是基于模块进行压力测试，还是发起全链路压测。

如果性能是背景数据敏感的场景，还需要确定背景数据量级与分布，并决定产生背景数据的技术方案，比如是通过 API 并发调用来产生测试数据，还是直接在数据库上做批量 insert 和 update 操作，或者是两种方式的结合。

最后，无论采用哪种方式，都需要明确待开发的单用户脚本的数量，以便后续能够顺利组装压测测试场景。

性能测试的实施，是一个比较复杂的问题。首先，需要根据你想要解决的问题，确定性能测试的类型；然后，根据具体的性能测试类型开展测试。

1. 性能测试的实施，往往先要根据业务场景来决定需要开发哪些单用户脚本，脚本的开发会涉及到很多性能测试脚本特有的概念，比如思考时间、集合点、动态关联等等。
2. 脚本开发完成后，你还要以脚本为单位组织测试场景（Scenario），场景定义简单来说就是百分之多少的用户在做登录、百分之多少的用户在做查询、每个用户的操作步骤之间需要等待多少时间、并发用户的增速是 5 秒一个，还是 5 秒 2 个等等。
3. 最后，才是具体的测试场景执行。和自动化功能测试不同，性能测试执行完成后性能测试报告的解读，是整个测试过程中最关键的点。

如果你现在不太清楚我上面提到的一些概念也没关系，我会在后续的文章中为你详细讲解。

除了我给你详细分析的、最常用的功能测试、兼容性测试和性能测试外，还有很多测试类型（比如，接口测试、集成测试、安全测试、容量验证、安装测试、故障恢复测试等）。这些测试类型，都有各自的应用场景，也相应有独特的测试方法，在这里我就不再一一展开了，如果你有关于这些测试类型的问题，可以给我留言讨论。

测试资源

测试资源通常包括测试人员和测试环境，这两类资源都是有限的。测试计划的目的是，保证在有限资源下的产出最大化。所以，**测试资源就是需要明确“谁来测”和“在哪里测”这两个问题。**

测试人员是最重要的，直接关系到整个测试项目的成败和效率。测试人员的资源通常有两个维度：一是，测试工程师的数量；二是，测试工程师的个人经验和能力。

你会发现，测试工程师的经验和能力不足，很难通过测试人员的数量来弥补。相反地，测试工程师的经验和能力都非常强的情况下，测试人员的数量可以适当地减少。

通常在测试团队中，测试工程师既有资深，也会有初级，那么你就必须针对团队的实际情况去安排测试计划。比如，难度较大的工作，或者一些新工具、新方法的应用，又或者自动化测试开发工作，通常由资深的测试工程师来承担；而那些相对机械性、难度较小的工作，则由初级工程师完成。

可见，你要想规划好测试资源，除了要了解项目本身外，还必须对测试团队的人员特点有清晰的把控。另外，我强烈建议你具体的任务清晰地落实到每个人的身上，这将有利于建立清晰的责任机制，避免后续可能发生的扯皮。

相对于测试人员，测试环境就比较好理解了。不同的项目，可能会使用共享的测试环境，也可能使用专用的测试环境，甚至还会直接使用生产环境。另外，对于目前一些已经实现容器化部署与发布的项目，测试环境就会变得更简单与轻量级，这部分内容我会在后续的文章中给你详细讲解。

测试进度

在明确了测试范围、测试策略和测试资源之后，你就要考虑具体的测试进度了。**测试进度主要描述各类测试的开始时间，所需工作量，预计完成时间，并以此为依据来建议最终产品的上线发布时间。**

比如，版本接受测试（Build Acceptance Test）的工作量，冒烟测试（Smoke Test）的工作量，自动化脚本开发的工作量，缺陷修复的验证工作量，需要几轮回归测试、每一轮回归测试的工作量等等。

在传统瀑布模型中，测试进度完全依赖于开发完成并递交测试版本的时间。如果开发提交测试版本发生了延误，那么在不裁剪测试需求的情况下，产品整体的上线时间就同样会延期。

然而在敏捷模式下，测试活动贯穿于整个开发过程，很多测试工作会和开发工作同步进行，比如采用行为驱动开发（Behavior-Driven Development）模式，这样测试进度就不会完全依赖于开发递交可测试版本的时间。

行为驱动开发，就是平时我们经常说的 BDD，指的是可以通过自然语言书写非程序员可读的测试用例，并通过 StepDef 来关联基于自然语言的步骤描述和具体的业务操作，最典型的框架就是知名 “Cucumber” 。

测试风险预估

俗话说，计划赶不上变化，对于测试也是一样的道理，很少有整个测试过程是完全按照原本测试计划执行的。通常需求变更、开发延期、发现重大缺陷和人员变动是引入项目测试风险的主要原因。

对于需求变更，比如增加需求、删减需求、修改需求等，一定要重新进行测试需求分析，确定变更后的测试范围和资源评估，并与项目经理和产品经理及时沟通因此引起的测试进度变化。测试经理 / 测试负责人切忌不能有自己的咬牙扛过去的想法，否则无论是对测试团队还是对产品本身都不会有任何好处。

另外，随着测试的开展，你可能会发现前期对于测试工作量的预估不够准确，也可能发现需要增加更多的测试类型，也可能发现因为要修改测试架构的严重缺陷而导致很多的测试需要全回归，还有可能出现开发递交测试版本延期，或者人员变动等各种情况。

所以，**在制定测试计划时，你就要预估整个测试过程中可能存在的潜在风险，以及当这些风险发生时的应对策略。** 那么，在真的遇到类似问题时，你才可以做到心中不慌，有条不紊地应对这些挑战。

总结

软件测试同软件项目一样，也要制定详细的测试计划。虽然在敏捷开发模式下，软件测试不再局限于厚重的、正式的计划文档，但是测试计划的重要性丝毫没有发生变化。

一份成功的测试计划，必须清楚地描述：测试范围、测试策略、测试资源、测试进度和测试风险预估这五个最重要的方面。

测试范围需要明确 “测什么” 和 “不测什么”；测试策略需要明确 “先测什么后测什么” 和 “如何来测”；测试资源需要明确 “谁来测” 和 “在哪里测”；测试进度是需要明确各类测试的开始时间，所需工作量和预计完成时间；测试风险预估是需要明确如何有效应对各种潜在的变化。

思考题

在这篇文章中，我只和你分享了做好测试计划中最最关键的内容，意在抛砖引玉。那么，你在工程实践中，还有哪些见解呢？

欢迎你给我留言。

 极客时间

软件测试52讲

从小工到专家的实战心法



茹炳晟 eBay中国研发中心
测试基础架构技术主管

新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 07 | 如何高效填写软件缺陷报告？

下一篇 09 | 软件测试工程师的核心竞争力是什么？

精选留言 (42)

写留言



Robert小七

2018-07-27

14

目前测试都是和开发一起来评估工作量，但是开发的工作量可以根据静态页面数量和接口数量，而测试的工作量很难现场评估，那么老师在工作中是如何有效评估测试工作量的？

展开



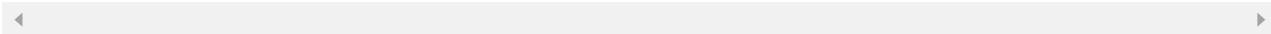
余衫酉

2018-07-16

👍 13

有开发伙伴在么？以测试的角度反观开发，收获也不少。可以预见性地减少bug。

作者回复: 非常同意你的观点，这其实也是我想传达的信息👍



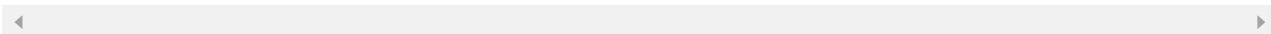
Mr.Z

2018-07-16

👍 10

我们在快速迭代的逻辑业务型版本项目中建议增加产品需求测试，意义在在开发和测试开发前，保证所有人对需求理解的一致性。

作者回复: 很好的办法，需求是一切的源头



subona

2018-10-11

👍 3

我们公司很多项目的迭代过程都是先定好发版时间，然后除去开发预估时间，剩下就是测试的时间了



口水窝

2019-03-06

👍 2

以前待得一家公司有过测试计划，那时候感觉是流于形式，现在仔细回忆起来，也就是哪家公司，教会了我测试规范、测试思想一些最初的概念、模型。现在待得小公司，就我一个测试，没有测试计划，但是主要做了功能测试，其余的根据自己的时间、进度来安排，后面还是主要精力放在自动化测试、性能测试的把握上，给自己在测试领域的范围上提升一个高度！

展开 ∨



小老鼠

2018-10-08

👍 2

敏捷测试计划与传统测试计划区别，在敏捷测试中测试计划是否与开发计划合并？如何作好测试需求？

没有性能测试需求如何作性能测试？

性能测试范围如何确定？

对于需要几轮回归我建议在测试计划中不要太具体，这个变数太大了。

展开 ▾



Knight

2018-07-18

👍 2

做好测试计划 很多时候被风险给打乱，也就是不确定性打乱。

如 跨部门合作，技术攻关 预研项目不成熟等的引进等的不确定性

同时测试计划受开发进度 质量影响很大，可能之前估计1周搞完的结果质量太差 需要搞两周 那最后的发布日期都定下来 很可能就堆人，压缩测试周期了



天天向

2018-07-16

👍 2

BDD模式不是很明白，后续有详细介绍吗？

展开 ▾



syland215

2019-02-12

👍 1

相对来说，我对测试策略比较关注，好的测试策略真的能起到事半功倍的效果。

打个简单的比方吧，一个主干业务的用例，早期执行发现问题的修复成本比晚期发现问题的修复成本要低太多太多，这里说的修复成本也包含测试投入的成本，这个相信有一定测试经验的同学肯定都深有体会。...

展开 ▾



阿甘

2018-07-20

👍 1

学到东西了，但是公司开发和测试的时间压的很短，老师有什么办法解决吗，因为可能会造成生产上存在bug，原因是否因为回归的不充分导致的呢？有什么好的解决方案吗

展开 ▾



秋荣

2018-07-19

👍 1

在您的项目中，测试计划真的会写那么详尽吗？是以word文档形式存在吗？是每个需求去编写吗？我们团队使用敏捷模式，基本很少有测试计划，想跟您取取经

展开 ∨



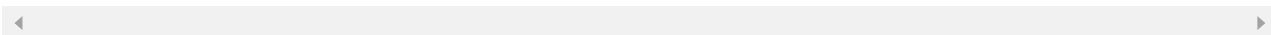
kilyun

2018-07-17

👍 1

对于测试计划在项目过程中的变化，我们公司的做法是不会去更新测试计划文档，只会通过邮件等知会到相关人，这样一来的话等项目结束了再翻看测试计划实际很多信息都是不正确的了，也是没有参考价值了，经老师讲解的测试计划的重要性，要实现测试计划的落地还有很长一段路要走...

作者回复: 希望文章中提到的方法能够对你们有帮助，另外，其实我们并不强求一定要文档化的测试计划，测试计划最核心的作用是帮助梳理测试过程的各个方面，测试计划的本质还是手段，而不是目的。



kilyun

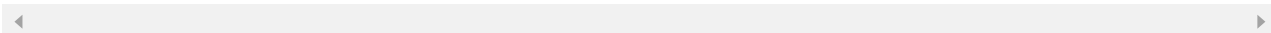
2018-07-17

👍 1

老师，请教下，对于兼容性漏测的问题如何快速定位原因，如用户使用我公司的APP反馈某款android机型出现了一个问题，其他手机上没法重现此问题，这种情况下该如何来处理该问题，谢谢~

展开 ∨

作者回复: 比较推荐的做法是根据大数据统计结果找到top 30%的机型，android版本，还有机型分辨率，然后优先覆盖这部分单位兼容性测试，往往一些典型的兼容性问题都可以在这里被发现，对于那些只在小版本或者非主流机型上特有的兼容性问题，并没有更好的方法，但是可以通过完整的后台log来帮助收集相关信息。



雪落无声

2018-07-16

👍 1

目前AI 领域内的测试，确定具体的测试数据集以及数据集的来源也是在制定测试计划时很重要的一个模块



橄榄



2018-07-16



打卡08

展开 ∨

作者回复: 期待继续打卡



泡芙小姐

2019-05-23



我们公司在项目开发前项目经理会给开发和测试预定开发时间和测试时间以及上线时间，我们在测试的过程中会遇到开发延期的问题，比如开发延期部署回归环境，开发在测试中修改一个比较难的bug,项目最终会延期，但是项目经理都认为是测试延期，因为他只看测试完成的最后时间。但是其实这些都不是测试的原因啊，请问该如何处理这种情况

展开 ∨



Geek_d3aec...

2019-04-10



目前 做测试项目，看完 茹老师，专业的测试计划 包括：1.测试范围，测什么，不测什么，2.测试策略，先测什么，后测什么 3.如何来测 4.测试资源，谁来测试，在哪里进行测试 5.测试进度，现学现用。

展开 ∨



妮妮王

2019-03-25



一般初期产品比较适合做计划吧，中间迭代的优化点，没啥必要吧，毕竟实时在更新



fun

2018-12-03



深有感触，在测试执行过程中经常存在思维惰性和侥幸心理，不愿对"变化"做出测试计划的调整。需求变化时千万不能以为自己能咬牙扛过去。不仅仅需求变更，技术方案调整，都应该仔细评估和领导层上报风险，这个道理之前没有彻底的觉悟，如果用一个失败的项目换教训。要是早点听到老师的课并好好揣摩理解就好了

展开 ∨

作者回复: 能够有感悟和收获，我也觉得很开心



ceshi小白

2018-12-03



从第一篇看到这里，还一直有想看下篇的欲望，老师讲的内容不仅适合测试小白，更能让我们真正觉得很实用，不会那么宽泛，或者看着看着就打瞌睡了，希望老师多出一些关于测试的课程，让我们学到更多测试技能!!!

展开 ∨