

第5章 软件评审

在软件生命周期的某一阶段中出现的缺陷,如果得不到及时的纠正,就会传播到开发的后续阶段中,并在后续阶段中引出更多的缺陷,纠正的成本会显著增加。实践证明,提交给测试阶段的产品中包含的缺陷越多,经过同样时间的测试之后,产品中仍然潜伏的缺陷也越多。所以必须将发现缺陷的工作提前,在开发时期的每个阶段,都要进行严格的软件评审,尽量不让缺陷带到下一阶段。本章分别论述了软件评审的含义和软件评审的原则,软件评审的阶段,以及如何避免进入评审误区,软件评审中的角色和职能。

5.1 软件评审概述

为什么需要软件评审?总体来说,在开发过程中,评审可以获得以下收益。

- 提高项目的生产率。这是由于早期发现了错误,因而减少了返工时间,还可能减少测试时间。
- 改善软件的质量。
- 在评审过程中,使开发团队的其他成员更熟悉产品和开发过程。
- 通过评审,标志着软件开发的一个阶段的完成。

软件评审能生产出更容易维护的软件。主要原因:对于被评审的软件,评审者必须是非常熟悉的;同时,在评审过程中,一定会产生并利用很多证明文档,于是评审就迫使开发者产生许多有用的文档,而这些文档如果不是因为评审,在整个项目期间可能都不会生产。此外,评审过程也将增加对所开发软件的理解。

经验表明,在制定技术规范期间产生的问题如果在集成测试或产品使用时被发现,与在设计或编码期间被发现相比较,返工的成本前者要比后者高 10~100 倍。软件评审是对软件元素或者项目状态的一种评估手段,以确定其是否与计划的结果保持一致,并使其得到改进。

1994 年,IEEE 对软件评审下的定义:软件评审是一种对软件元素所做的正式的评审活动。其目的是检验软件开发和软件测试各阶段的工作是否齐全、规范,各阶段产品是否达到了规定的技术要求和质量要求,以决定是否可以转入下一阶段的工作。

M.E. Fagan 在软件评审方面有突出的贡献,他在总结大量的实践后得到的结论是,用人们熟悉的运行程序的测试方法只能发现 1/5 的故障,而认真的评审可以发现 4/5 的故障。

Karl E. Wieggers 对软件评审的阐述:不管你有没有发现它们,缺陷总存在,问题只是你最终发现它们时,需要多少纠正成本,评审的投入把质量成本从昂贵的、后期返工转变为早期的缺陷发现。

5.2 软件评审的主要内容

软件评审并不是在软件开发完毕后进行评审,而是在软件开发的各阶段都要进行评审。因为在软件开发的各阶段都可能产生错误,如果这些错误不能被及时发现并纠正,会不断扩大,最后可能导致开发失败。

软件评审的主要内容包括软件评审的目标、软件评审的过程、软件评审的原则和软件评审的特点。

5.2.1 软件评审的目标

软件评审的目标主要如下。

- (1) 发现任何形式表现的软件功能、逻辑或实现方面的错误。
- (2) 通过评审验证软件的需求。
- (3) 保证软件按预先定义的标准表示。
- (4) 已获得的软件是以统一的方式开发的。
- (5) 使项目更容易管理。

5.2.2 软件评审的过程

软件评审的过程如下。

(1) 召开评审会议:一般应有3~5人参加,会前每个参加者做好准备,评审会每次一般不超过2小时。

(2) 会议结束时必须做出以下决策之一:接受该产品,不需做修改;由于错误严重,拒绝接受;暂时接受该产品。

(3) 评审报告与记录:所提出的问题都要进行记录,在评审会结束前产生一个评审问题表,另外必须完成评审简要报告。

5.2.3 软件评审的原则

软件评审是相当重要的工作,也是目前国内开发最不重视的工作。软件评审的原则如下。

- (1) 评审产品,而不是评审设计者(不能使设计者有任何压力)。
- (2) 会场要有良好的气氛。
- (3) 建立议事日程并维持它(会议不能脱离主题)。
- (4) 限制争论与反驳(评审会不是为了解决问题,而是为了发现问题)。
- (5) 指明问题范围,而不是解决提到的问题。
- (6) 展示记录(最好有黑板,将问题随时写在黑板上)。

- (7) 限制会议人数和坚持会前准备工作。
- (8) 对每个被评审的产品列出评审清单(帮助评审人员思考)。
- (9) 对每个正式技术评审分配资源和时间进度表。
- (10) 对全部评审人员进行必要的培训。
- (11) 及早地对自己的评审做评审(对评审准则的评审)。

5.2.4 软件评审的特点

软件评审的特点如下。

- (1) 发现隐藏的软件缺陷。
- (2) 参加评审的人员应以软件项目开发组以外的同行人员为主,如果项目组人员参加也是起帮助理解评审对象的作用。
- (3) 被评审的对象通常指的是软件开发中的各种技术产品,如需求规格说明、用户手册、测试计划等。
- (4) 评审有多种形式,主要有正式评审和非正式评审。其中,非正式评审也可以用相关人员分散阅读,以书面意见的方式进行,但必须有评审人员的签名,以体现评审的责任。

5.3 软件评审的阶段

软件评审的阶段包括需求评审、概要设计评审、详细设计评审、数据库设计评审和测试评审等。

5.3.1 需求评审

1. 需求评审概述

软件需求是软件开发的最重要的一个步骤,需求的质量很大程度上决定了项目质量或产品质量。需求风险也常常是软件开发过程中最大的一个风险,降低需求风险的一个重要手段就是需求评审,但是需求评审是所有的评审活动中最难的一个,也是最容易被忽视的一个评审。

需求评审中经常存在以下问题:

- 需求报告很长,短时间内评审者根本不能把需求报告读懂,想清楚;
- 没有做好前期准备工作,需求评审的效率很低;
- 需求评审的节奏无法控制;
- 找不到合格的评审员,与会的评审员无法提出深入的问题。

2. 几种失败的需求评审

案例一:某软件公司内部举行产品的需求评审会,主要是公司内部的领域专家参加,

在评审会开始后不久,某领域专家就对需求报告中的某个具体问题提出了自己的不同意见,于是,与会人员纷纷就该问题发表自己的意见,大家争执不下,结果致使会议出现了混乱状况,主持人无法控制局面,会议大大超出了计划评审时间。

案例二:某软件公司为某公司 A 做业务流程管理系统的需求评审会,当项目组人员在会议上宣读多达上百页的需求报告时,用户明确提出听不懂,致使会议不得不改日进行。

案例三:某软件公司在用户处开完物资管理系统的需求评审会后,与会人员离开会议室时,纷纷摇头,认为本次会议没有多少实际效果,完全是在走过场。

3. 如何做好需求评审

1) 分层次评审

用户的需求是可以分层次的,一般而言可以分成如下层次。

- (1) 目标性需求:定义了整个系统需要达到的目标。
- (2) 功能性需求:定义了整个系统必须完成的任务。
- (3) 操作性需求:定义了完成每个任务的具体的人机交互。

目标性需求是企业的高层管理人员所关注的,功能性需求是企业的中层管理人员所关注的,操作性需求是企业的具体操作人员所关注的。对不同层次的需求,其描述形式是有区别的,参与评审的人员也是不同的。如果让具体的操作人员去评审目标性需求,可能会很容易地导致捡了芝麻、丢了西瓜的现象,如果让高层的管理人员也去评审那些操作性需求,无疑是一种资源的浪费或者就会出现案例三的情形。

2) 正式评审与非正式评审结合

正式评审是指通过开评审会的形式,组织多个专家,将需求涉及的人员集合在一起,并定义好参与评审人员的角色和职责,对需求进行正规的会议评审。而非正式评审并没有这种严格的组织形式,一般也不需要将人员集中在一起评审,而是通过电子邮件、文件汇签甚至是网络聊天等多种形式对需求进行评审。两种形式各有利弊,但往往非正式的评审比正式的评审效率更高,更容易发现问题。因此在评审时,应该更灵活地利用这两种方式。

3) 分阶段评审

应该在需求形成的过程中进行分阶段的评审,而不是在需求最终形成后再进行评审。分阶段评审可以将原本需要进行的大规模评审拆分成各个小规模评审,降低了需求返工的风险,提高了评审的质量。例如,可以在形成目标性需求后进行一次评审,在形成系统的初次概要需求后进行一次评审,当对概要需求细分成几部分时,对各部分进行评审,最终再对整体的需求进行评审。

4) 精心挑选评审员

需求评审可能涉及的人员包括需方的高层管理人员、中层管理人员、具体操作人员、IT主管、采购主管;供方的市场人员、需求分析人员、设计人员、测试人员、质量保证人员、实施人员、项目经理以及第三方的领域专家等。由于这些人员中所处的立场不同,对同一个问题的看法是不相同的,有些观点是和系统的目标有关系的,有些则关系不大,不同的

观点可能形成互补的关系。为了保证评审的质量和效率,需要精心挑选评审员。首先要保证使不同类型的人员都要参与进来,否则很可能会漏掉很重要的需求。其次在不同类型的人员中要选择那些真正和系统相关的,对系统有足够了解的人员参与进来,否则很可能使评审的效率降低或者最终不切实际地修改了系统的范围。

5) 对评审员进行培训

在很多情况下,评审员是领域专家而不是进行评审活动的专家,他们没有掌握进行评审的方法、技巧、过程等,因此需要对评审员进行培训,同样对于主持评审的管理者也需要进行培训,以便于参与评审的人员能够紧紧围绕评审的目标来进行,能够控制评审活动的节奏,提高评审效率,避免发生案例一和案例二中出现的现象。对评审员的培训也可以分为简单培训与详细培训两种。简单培训可能需要十几分钟或者几十分钟,需要将在评审过程中的需要把握的基本原则,需要注意的常见问题说清楚。详细培训则可能需要对评审的方法、技巧、过程进行正式的培训,需要花费较长的时间。详细培训是一个独立的活。需要注意的是,被评审人员也要被培训。

6) 充分利用需求检查单

需求检查单是很好的评审工具。需求检查单可以分成两类:需求形式的检查单和需求内容的检查单。需求形式的检查主要是针对需求文档的格式是否符合质量标准来提出的;需求内容的检查是由评审员负责的,主要是检查需求内容是否达到了系统目标、是否有遗漏、是否有错误等,这是需求评审的重点。检查单可以帮助评审员系统全面地发现需求中的问题,检查单也是随着工程财富的积累逐渐丰富和优化的。

7) 建立标准的评审流程

对正规的需求评审,需要建立正规的需求评审流程,按照流程中定义的活动进行规范的评审过程。例如,在评审流程定义中可能规定评审的进入条件,评审需要提交的资料,每次评审会议的人员职责分配,评审的具体步骤,评审通过的条件等。

8) 做好评审后的跟踪工作

在需求评审后,需要根据评审人员提出的问题进行评价,以确定哪些问题是必须纠正的,哪些可以不纠正,并给出充分的客观的理由与证据。当确定需要纠正的问题后,要形成书面的需求变更的申请,进入需求变更的管理流程,并确保变更的执行,在变更完成后,要进行复审。切忌评审完毕后,没有对问题进行跟踪,而无法保证评审结果的落实,使前期的评审努力付之东流。

9) 充分准备评审

评审质量的好坏很大程度上取决于在评审会议前的准备活动。常出现的问题是,需求文档在评审会议前并没有提前下发给参与评审会议的人员,没有留出更多、更充分的时间让参与评审的人员阅读需求文档。更有甚者,没有执行需求评审的进入条件,在评审文档中存在大量的低级的错误或者没有在评审前进行沟通,文档中存在方向性的错误,从而导致评审的效率很低,质量很差。对评审的准备工作,也应当定义一个检查单,在评审之前对照检查单落实每项准备工作。

5.3.2 概要设计评审

在软件概要设计结束后必须进行概要设计评审,以评价软件设计说明书中所描述的软件概要设计在总体结构、外部接口、主要部件功能分配、全局数据结构以及各主要部件之间的接口等方面的合适性。

一般应考察以下五方面:

- (1) 概要设计说明书是否与软件需求说明书的要求一致;
- (2) 概要设计说明书是否正确、完整和一致;
- (3) 系统的模块划分是否合理;
- (4) 接口定义是否明确;
- (5) 文档是否符合有关标准规定。

5.3.3 详细设计评审

在软件详细设计阶段结束后必须进行详细设计评审,以评价软件验证与确认计划中规定的验证与确认方法的合适性与完整性。

一般应考察以下五方面:

- (1) 详细设计说明书是否与概要设计说明书的要求一致;
- (2) 模块内部逻辑结构是否合理,模块之间的接口是否清晰;
- (3) 数据库设计说明书是否完全,是否正确反映详细设计说明书的要求;
- (4) 测试是否全面、合理;
- (5) 文档是否符合有关标准规定。

5.3.4 数据库设计评审

在数据库设计阶段结束后必须进行数据库设计评审,以评价数据库的结构设计及运用设计的合适性。一般应考察以下五方面:

- (1) 概念结构设计;
- (2) 逻辑结构设计;
- (3) 物理结构设计;
- (4) 数据字典设计;
- (5) 安全保密设计。

5.3.5 测试评审

测试评审主要对测试的各环节进行评审,具体内容如下:

- (1) 软件测试需求规格说明的评审;

- (2) 软件测试计划的评审；
- (3) 软件测试说明的评审；
- (4) 软件测试报告的评审；
- (5) 软件测试记录的评审。

5.4 避免进入评审误区

误区一：评审参与者不了解评审过程。

如果评审参与者不了解整个的评审过程，就会有一种自然的抗拒情绪，因为大家看不到做这件事情的效果，感觉到很迷茫，这样会严重影响大家参与评审的积极性。

误区二：评审人员评论开发人员，而不是产品。

评审的主要目的是发现产品中的问题，而不是根据产品来评价开发人员的水平。但是往往会出现把产品质量和开发人员水平联系起来的事情，于是评审变成了批斗大会，极大地打击了开发人员的自尊心，以致严重地影响了评审的效果。

误区三：评审没有被安排进入项目计划。

参与评审需要投入大量的时间和精力，应该被安排进入项目计划中。但是现实的情况往往是评审变成了义务劳动，参与评审的人员必须加班加点才能完成评审任务。如此一来，出现评审人员对评审对象不了解的情况也就不足为奇了。

误区四：评审会议变成了问题解决方案讨论会。

评审会议主要的目的是发现问题，而不是解决问题，问题的解决是评审会议之后需要做的事情。但是，由于开发人员对技术的追求，评审会议往往变成了问题研讨会，大量地占用了评审会议的时间，导致大量评审内容被忽略，留下无数的隐患。

误区五：评审人员事先对评审材料没有足够的了解。

任何一份评审材料都是他人智慧和心血的结晶，需要花足够的时间去了解、熟悉和思考。只有这样，才能在评审会议上发现有价值的深层次问题。在很多的评审中，评审人员因为各种原因，在评审会议之前对评审材料没有足够的了解，于是出现了评审会议变成了技术报告的怪现象。

误区六：评审人员关注非实质性问题。

经常会出现这样的问题：在评审中，评审人员过多关注一些非实质性的问题，例如文档的格式、措辞，而不是产品的设计。出现这种情况可能的原因：没有选择合适的人参加评审；评审人员对评审对象没有足够的了解，无法发现深层次的问题。

误区七：忽视细节。

在组织评审的过程中，很多人不太注意细节。例如，会议时间的设定、会议的通知、会议场所的选择、会场环境的布置、会议设施的提供、会议上气氛的调节和控制等，而实际上这样的细节会大大影响评审会议的效果。很难想象，大家在一个空气混浊、噪声很大的会议室里面能够全身心地投入工作。

其他误区还有：

- 人身攻击。在评审过程中，所有的参与人都应该将矛盾集中于评审内容本身，而

不能针对特定的参与人。

- 无休止的争论。通常对于某些问题,评审组很难达成一致意见,这时,可以把问题记录下来,而如何认定则留给作者自己决定。
- 偏离会议中心。在实际会议中,会议常常会发生偏离,如转到政治话题的讨论。
- 鼓励所有人发言。鼓励不善言辞的参与者就评审内容发表自己的看法,例如按照座位顺序轮流发表意见。

5.5 软件评审中的角色和职能

在正式评审中,每个参加者分别担当不同的角色,完成各自的职责。一个评审小组由3~5人组成,主要成员有评审组长、宣读员、记录员、评审员等。

1. 评审组长

评审组长(Review Team Leader)在整个评审会议中起领导作用,负责组织、主持和控制软件评审活动。他的主要任务如下。

- 决定软件评审人员,并明确各自的角色和职能。
- 安排正式的软件评审会议。
- 安排软件评审的内容。
- 确保所有提出的缺陷都被记录在案。
- 跟踪问题的解决情况。
- 与项目负责人沟通评审结果。

2. 宣读员

宣读员(Reader)的主要任务是在评审会议上引导评审小组阅读被审资料。

3. 记录员

记录员(Recorder)的主要任务是将评审会议上发现的问题记录在“软件评审问题记录表”上。

4. 评审员

评审员(Reviewer/Inspector)的主要职责如下。

- 熟悉评审内容,为评审做好准备。
- 在评审会议上应该关注问题而不是针对个人。
- 主要的问题和次要的问题可以被分别讨论。
- 在会议前或者会议后可以就存在的问题提出建设性的意见和建议。
- 明确自己的角色和责任。
- 做好接受错误的准备。

5.6 思考题

1. 为什么需要软件评审？或者说评审可以获得哪些收益？
2. 软件评审工作中存在哪些误区？
3. 简述软件评审的阶段。
4. 正式评审与非正式评审有何区别？
5. 简述概要设计评审和详细设计评审。
6. 简述软件评审中的角色和职能。

