

第1章

业务质量保障介绍



1.1 电商行业项目的特点

近年来，中国电子商务蓬勃发展，电子商务与日常百姓生活关系越来越密切，购物消费、上门服务、订购机票、预订酒店、旅游门票、手机充值等覆盖了生活的方方面面。

笔者一直在电子商务类公司做软件质量团队的相关管理工作，参与了不少电商项目的测试。下面笔者根据观察到的一些现象，站在测试的角度对电商行业项目的特点进行如下总结。

(1) 注重易用性

电子商务作为市场热门行业，激烈的市场竞争是必须面对的事实。大家都想吸引更多的用户，对于用户而言方便易用的软件更受欢迎。软件的易用性是软件质量中重要的特性，简单地讲就是用户在使用过程中是否感觉方便舒适。在 2003 年颁布的 GB/T16260-2003 (ISO 9126-2001)《软件工程产品质量》质量模型中，提出易用性包含易理解性、易学习性和易操作性。关于易用性，在具体的项目测试中我们通常首先要关注 UI (用户界面, User Interface)。用户在接触软件时首先看到的就是 UI，为了给用户留下良好的第一印象，重要系统都有自己的 UI 规范，其定义了界面细致的标准，如导航、表单、表格、标签、按钮、翻页、进度条、文字错误提示等。除了符合规范，我们还要关注是否舒适直观、洁净、不唐突。易用性方面还要关注产品功能是否容易理解，操作是否简单不繁杂等。在实际项目中，这些内容往往是产品部门的同事做的工作比较多，但是作为测试工程师在测试过程中要有这方面的意识。

(2) 注重安全性

据统计，电商企业在营销过程中，有 90% 的客户会因为缺乏安全感和对网络信任度低而流失，安全问题是电子商务的核心问题。据 2016 年京东财报显示，京东年度活跃用户数已经达到 2.266 亿，如此庞大的用户量如果安全没有保障，则将带来巨大的损失。京东有专门的网络安全团队，对重要系统上线前都会进行安全扫描，也会定期对线上运行的系统进行安全检查。如果发现有安全漏洞，则会第一时间联系对应系统负责人解决处理。但是京东的系统 and 业务数量非常庞大，有些业务上的

安全问题全部依靠有限的安全部门同事检查并不够。所以在各系统功能测试阶段，测试工程师需要首先进行一轮安全测试。

(3) 项目周期短，需求变化快

在传统的软件项目中，需求一般是十分确定的，交付时只要与需求一致即可，往往采用瀑布模式，项目周期相对较长。电子商务行业作为市场热门行业，激烈的市场竞争是必须面对的事实。电子商务项目相比传统项目需求变化快，当一个需求提出后，如果项目周期太长，那么交付时可能已经不适用当前的环境了。所以一个需求或项目从提出到上线交付根据功能复杂程度一般周期为几天到几个月。作为测试工程师在保障软件质量的同时应尽力通过技术和工作方法提高效率，缩短测试时间从而缩短整个项目的周期。

1.2 测试流程中的重点工作

在京东研发体系内部，由于业务差异性，工作流程也不尽相同。不同的部门根据自身的情况和业务内容会制定适合自己的工作流程。笔者以 POP（Platform Open Plan，开放平台）平台部门的测试工作流程为例向大家介绍工作流程中需注意的事项，供大家参考。POP 平台涉及的业务由多条业务线组成，每条业务线由产品经理、开发工程师和测试工程师组成。

1.2.1 测试前的工作

1. 排期沟通

首先介绍一下小需求和项目的概念。小需求一般指功能较少，研发工期较短的任务。项目是相对于小需求而言功能较多，涉及的系统也较多，对应研发工期较长的任务。实际上项目也是由多个不同的小需求组成的。每周五，业务线中的产品经理会联系对应的开发和测试人员到一起，沟通需求排期情况。在这个沟通会议中，产品经理首先会列出上一周进入开发阶段的需求列表和未来（一般 3 个月内）要做但未排期的需求列表。将上周进入开发阶段需求列表中的需求进行状态更新，去除已经上线完成的需求并加入本周新进入开发阶段的需求。如果有新未排期的需求，则更新到排期需求列表中。在此过程中，产品经理、开发工程师、测试工程师一起

沟通需求进度和风险，是否需要协调资源，并对未来一段时间的工作内容有大致的了解。在沟通会结束后，产品同事会把最新的表格作为邮件附件发送给业务线成员。

2. 提测需求列表

京东有个研发知识管理平台（RDKM），在 RDKM 中创建了以周为单位的提测需求列表。表中包括需求名称、需求介绍、文档链接、开发、产品、上线日期、所属条线、提测时间、测试开始时间、测试工程师。如图 1.2.1 所示。

| 序号 | 需求名称 | 需求介绍 | 文档链接 | 开发 | 产品 | 上线日期 | 所属条线 | 提测时间 | 测试开始时间 | 测试工程师 |
|----|------|------|------|----|----|------|------|------|--------|-------|
| 1 | | | | | | | | | | |
| 2 | | | | | | | | | | |
| 3 | | | | | | | | | | |
| 4 | | | | | | | | | | |
| 5 | | | | | | | | | | |
| 6 | | | | | | | | | | |
| 7 | | | | | | | | | | |
| 7 | | | | | | | | | | |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | |
| 10 | | | | | | | | | | |

图 1.2.1

下面分别解释下该表格中各字段的含义。

- 需求名称：对需求功能的简单描述，如果是项目，则此处是项目名称；
- 需求介绍：对需求内容进一步概括总结；
- 文档链接：产品的需求文档及开发的设计文档链接；
- 开发：实现需求的开发工程师，有程序问题可以直接找到本人；
- 产品：提出需求的产品经理，有需求问题可以直接找到本人；
- 上线日期：预计需求完成测试并上线的时间；
- 所属条线：在业务线内部进行了更为细致的划分，每个业务线的测试工程师对应一个或几个条线；
- 提测时间：开发完成后可以提交测试的时间；
- 测试开始时间：测试工程师根据需求内容及自身计划，开始测试的时间；
- 测试工程师：需求对应的测试工程师。

这个文档每周由开发和测试工程师维护。每周五前开发工程师会把下周要提交测试的需求列入对应的 RDKM 链接中，填写内容包括需求名称、需求介绍、文档链接、所属条线、上线时间、产品、开发、提测时间。测试工程师根据需求内容，认领属于自己的工作内容并填写测试开始时间及姓名。如果遇到本周没有完成的需求，那么由测试工程师在周五下班前将该需求信息复制到下周的提测需求列表中。

这样每个测试工程师都能提前知道下周的具体工作内容。如果遇到资源问题也能提前准备，如提交测试时间与上线时间太近、测试时间不够等。虽然在前一周就知道了本周的测试内容，但是有些紧急需求是预想不到的。如线上有问题需要修改，优先级自然最高。如果能够保证其他需求正常完成最好，如果影响到其他需求的进度，那么测试工程师将与产品经理及开发工程师沟通，保障优先级高的需求不受影响。

3. 项目需求评审

如果全部都是功能少、工期短的小需求，那么以上的计划提前一周是可以准备的。实际上还有一些项目级别的需求，除了以上的流程还要多做一些工作。和大多数公司一样，项目由于功能较多，涉及系统较多，在需求完成后产品同事需要邀请开发和测试工程师进行评审。评审有以下几个目的：

- 让开发和测试工程师了解需求要做的内容，以便之后各自对开发阶段和测试阶段进行排期。
- 评审过程中业务线成员对需求内容的模糊点进行提问，减少对需求的误解和日后的沟通。
- 在评审的过程中可能会发现需求本身的问题，及时发现能防止日后产生雪球效应。
- 业务线成员在一起沟通，能对系统有对应的认识，更能补充完善需求。

4. 项目的设计评审

项目需求评审通过以后，开发工程师就可以做项目的系统设计了。设计文档包括实现整个项目的架构设计和接口设计等详细信息，完成后也需要评审，测试工程师参加设计评审的意义在于：

- 深入了解项目构成，可以把项目拆解成不同的功能模块。

- 在时间紧迫的情况下，可以按功能模块分工测试，提前完成测试工作。
- 准备测试用例时只看需求文档是不够的，通过需求文档只能覆盖业务上的功能，根据设计文档能够设计分支及异常测试数据流。

通过设计评审对系统有了较深入的认识后就可以对测试进行排期了。

5. 项目的排期

通过需求和设计的评审，业务线成员对项目都有了较为深入的认识，接下来就可以对各自负责的需求模块进行排期。排期主要包括任务的拆分及预估每个任务的起始和结束时间。根据实际需要测试排期主要分两种情况，一种是在开发整体完成后再进行测试，这样测试周期相对较短。因为省去了模块间不通时使用 Mock(模拟)的时间和后续模块间再次联调的时间。还有一种情况是在开发完成一个模块后就进入测试，这样整个项目周期较短。因为测试提前介入，等开发全部完成时，大部分的功能已经同步测试完，只需要把最后提交的模块测完再进行模块间联调即可。测试工程师选择哪种排期模式视项目情况而定，后续按照排期进行。

1.2.2 测试中的工作

1. 测试用例的编写

在需求和设计评审后，测试工程师就可以准备测试用例了。测试用例的设计与执行是软件测试的核心工作，涉及的方法有很多，验证点的粒度也不尽相同。笔者所在部门对于测试用例有对应的模板：**Actor**→业务对象→验证点。其中 **Actor** 是操作者，可以是使用系统的用户也可以是调用被测系统的其他系统。业务对象是指某具体的功能点，如上架商品、修改价格、订单出库、查询状态等。验证点就是对业务对象执行后产生效果的验证内容，包括数据库中的数据变化、页面变化、功能点等。

用例示例：商家新增商品，验证商品描述字段长度是否从 3000 放开到 5000，且数据库表正确写入该商品信息。

2. 需求变更通知

在较复杂的需求测试过程中，经常会出现需求变更现象。由于变更后的内容与原有内容差异较少，一般不会组织进行评审，从而导致经常出现产品经理、开发工

工程师、测试工程师之间信息不一致的情况，测试提出 bug 实际是由于需求变更未及时同步。为了避免需求变更导致信息不一致，笔者所在部门在流程上做了对应规范。首先，产品同事把变更后的需求文档上传至京东研发管理平台，系统会检测到变更，自动通过邮件发送最新需求文档给相关开发和测试工程师。然后开发工程师按照最新的需求文档修改设计文档，测试工程师修改测试用例。这样当需求变更时，很快就能使相关工程师了解最新需求。

3. 进度与风险通报

项目级别的需求进入测试阶段后，测试负责人要每日以邮件形式通报测试进度，发送给该项目的所有参与者。如图 1.2.2 所示是项目测试日报模板。

| XXX项目 | | | | | | | | | |
|-----------------|----|--------|--------|--------|-------|-------|------|--------|-----|
| 项目整体进度 | | 计划上线日期 | | | | 项目状态 | 正常 | 风险问题概述 | |
| 任务 | 进度 | 测试阶段 | 计划完成时间 | 实际完成时间 | 测试工程师 | 研发工程师 | 产品经理 | 项目经理 | 备注 |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| 待沟通解决问题（风险问题列表） | | | | | | | | | 责任人 |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| 上线日期历史变更原因 | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

图 1.2.2

日报中主要列举了项目中各模块当前的进度及问题。可以参考开始日期和完成日期判断进度是否有问题，项目中待沟通解决的问题都会列举对应责任人。这样，项目成员每天都能了解整个项目的状态并及时关注解决问题。

4. 分支环境与主干环境

由于多个开发工程师开发时一般采用并行开发的模式，为了减少冲突，会在多个分支上进行开发。开发工程师在分支环境上自测完成后，提交测试。测试工程师将分支代码部署到测试环境进行验证主流程的冒烟测试，如果没有问题则将进行全面的集成测试。测试通过后，开发工程师将分支代码合并到主干环境。测试工程师将在主干环境再次进行测试，测试通过后才能审批上线。

5. 产品试用机制

产品经理写的需求文档并不能把所有产品细节都描述清楚，开发工程师在设计时使用不同的实现方案可能会使产品体验有所不同。所以，开发完成后及时让产品经理试用确认很有必要。测试工程师在分支测试通过后，需邀请产品经理试用，并同步给相关开发工程师。产品经理对应用进行试用，查看是否符合自己的设计初衷。如果试用的结果与期望有较大差异，则提出相关问题通知开发工程师及时解决，无异议也必须给予确认。

6. 上线前检查

基于对线上问题的长期分析，我们发现，很多线上问题并不是由于代码逻辑特别复杂或者测试时没有考虑到所引起的，往往是上线时忽略了某些问题。所以上线前应该对经常出现问题的点做最后的检查，避免出现低级错误。下面几个问题可以参考：

- (1) 关联（所依赖或者被依赖）系统是否同步上线？
- (2) 关联系统上线计划是否已经相互告知并做好上线准备？
- (3) 紧急上线，是否测试充分？
- (4) 测试环境（测试数据）业务流是否存在不可测或部分不可测的问题？
- (5) 是否存在用例场景罕见或业务触发几率极低，是否暂未测试？
- (6) 测试环境和开发环境是否存在以变量直接赋 Hard Code（硬编码）值实现的应用？
- (7) 该项目或需求的帮助文档是否正确更新？
- (8) 测试环境和开发环境是否用条件编译开关？
- (9) 本业务需求程序设计是否需要做性能测试？

7. 测试报告

需求上线前的最后一步就是测试报告了，测试报告可以通过对测试结果和系统出现的缺陷分析得到对软件质量的评价。笔者所在部门为了节省编写测试报告的时间

间，把测试报告中通用的部分抽离出来，开发出自己的测试报告系统。测试工程师只要在系统上填写少量的信息就能形成要求的测试报告。如图 1.2.3 所示。

JIRA编号ddd 请填写需求id/项目id全称 搜索

欢迎您, Tester 【点此切换账号!】

+ 添加收件人

收件人

+ 添加抄送人

抄送

测试结果

bug分析: 本次测试共发现个bug, 已修改个, 未解决个, 延迟修改个, 挂起个, 重复BUG个, 无效BUG个, 不能重现个, 已完成个

风险分析: * 无风险 ○ 有风险

测试人员: Tester [保存] [取消]

测试开始: [] 测试结束: []

测试类型: 系统测试 浏览器: []

可测性评估:

- 是否有需求文档
- 是否有设计文档
- 是否经过代码评审
- 可测度是否100%
- 是否需要性能测试
- 是否需要安全测试
- 是否涉及DDL变更
- 是否符合UI规范
- 帮助文档是否检查通过

测试环境: IP: 192.168.102.14 [新增] domain: new.man.jd.net [新增] []

业务线: [] 涉及应用: [] git地址: [] [版本] [保存] [查看]

测试范围

测试设计: 无附件, 请到jira上传

BUG列表

| 序号 | bug描述 | 状态 | 报告人 | 解决 | 报告时间 | 解决时间 | bug链接 |
|-----------|-------|----|-----|----|------|------|-------|
| 没有找到匹配的记录 | | | | | | | |

打开新页面预览 使用您的邮件客户端预览 直接发送

图 1.2.3

1.2.3 上线后的工作

1. 线上验证

线上验证工作是产品经理、开发工程师、测试工程师都要做的。产品经理重点关注功能是否全部实现，UI 效果是否与期望一致。开发工程师关注上线后，是否有系统异常。测试工程师除了对一些由于特殊数据或特殊权限无法执行的测试用例不再执行验证，要尽可能把测试用例在线上全部执行验证一次。

2. 线上风险通报

测试工程师完成线上验证后，需要发送线上验证报告。线上验证报告中包括验证的需求名称、涉及上线人员、上线服务器 IP、测试用例及问题。有些重要的功能，如果在线上环境暂时无法验证，则要在报告中说明进行风险通报。这些问题，产品经理后续会联系业务人员或项目经理协调进行验证。

1.3 小结

不同行业、不同公司都有自己的测试流程，同一个公司不同部门的流程也是有差异的。笔者认为流程的制定需要参考业务的特点和要求。流程的作用是规范和提示工作步骤，要根据实际情况不断调整优化，以保障产品的质量。