

游戏软件功能测试——测试用例的编写方法浅谈

一、游戏软件与通用软件的区别

- a) 通用软件的需求明确, 游戏软件需求理想化
 - i. 通用软件中用户每步操作的预期结果都是明确且有规范可参考的, 而网游中并不是所有的需求都有一个明确的预期结果, 拿技能平衡性来说, 我们所谓的平衡也只是相对的平衡, 而非绝对的平衡。没有什么明确的参考参数。只能根据以往游戏的经验获得一个感知的结果。
 - ii. 网络游戏中的某些功能是有预期结果可参考的。例如组队、交易, 而另外一些带有策划创意的功能, 却是根据策划个人的理解, 来确定其预期结果的。人的思考力都是有限的, 所以不能保证在他的创意中会考虑到各种各样复杂的细节。也不能够保证这个创意就可以完全被用户所接受。

当你作为游戏测试人员时, 很多时候你需要做的不仅仅是验证功能。也需要帮助开发者和用户找到一个互相容忍的平衡点。游戏软件的测试员带有对策划需求的怀疑, 力求通过自己的努力在玩家和开发者之间将可能产生的矛盾减小。

- b) 通用软件开发过程中需求变更少, 游戏软件开发过程中需求变更更快
 - i. 通用软件的使用人群和软件的功能针对性, 决定软件从开始制作就很少再有新的需求变更。而游戏软件, 为了满足玩家对游戏的认可度, 策划需要不断的揣摩玩家的喜好, 进行游戏功能的改进。加之网游制作本身就是一个庞大复杂的工程, 开发者不可能做到在开发的前期, 就对游戏架构及扩展性做出最好的评估。所以导致为了满足用户的需求而不断的进行一些基础架构的修改, 基础架构的修改必然导致某些功能的颠覆。所以就出现了, 游戏开发过程中的一个恶性循环, 当基础架构修改到满意了, 玩家的需求又有了新的变化, 随之而来的又要进行新的调整, 再进行新的修改。最终导致了游戏软件的开发周期不断加长。任何一个有经验的团队, 对于每一个影响基础的改动都应该做出正确的评估。

二、网游有哪些测试内容

- a) 性能
 - i. 客户端性能
 - ii. 服务器端性能
 - 1. 服务器
 - 2. 数据库
 - iii. 网络
- b) 功能
 - i. 从运行完 `game.exe` 打开游戏界面后可进行的各种操作、玩法
 - ii. 界面
 - iii. 音乐
- c) 自动化
 - i. 测试工作组织实施中需要的工具、软件、平台的开发
 - ii. 自动化的回归测试作用: 游戏中基础的、变动不大的、出错率高的、可进行 `checklist` 重复测试的功能、性能等自动化是一个好方法
 - iii. 任何时候自动化都取代不了人脑, 它只是将一些重复性的劳动从我们测试人员身上去掉, 让我们有更多的时间做更有意义的事情, 如果你觉得你做一件事情是重复的, 且有规律可行的, 不妨考虑自动化

三、游戏中针对功能性测试测试用例编写浅谈

先了解下游戏中有哪些功能:

- a) 游戏发开中的功能有哪些
 - i. 不同的游戏对于功能的划分不同, 但是目前主流一些功能划分中有以下内容:
 1. 基础操作
 2. Npc
 3. 地图
 4. 装备
 5. 剧情
 6. 技能
 7. 人际
 8. PVP
 9.

这样我们很简单的将整个游戏的功能进行了划分, 划分完毕, 下来的工作就是针对某个功能的测试了。很多人都问过一个问题, 游戏测试中测试用例到底有什么用。下面继续~

- b) 游戏测试的测试用例有什么作用
 - i. 测试执行过程中, 按照用例指示的操作检查操作结果是否正确, 记录测试过程中发现的 bug
 - ii. 按照用例的执行结果确认功能的通过与否, 也有的按照用例的覆盖率来确定单服测试的通过与否
 - iii. 便于回归测试的执行

这样讲应该比较明白了吧。

- c) 测试用例应该包括什么——测试执行过程中所需的所有信息, 举例说明下。例如:
 - i. 表头: 功能名称、案例编写人员、编写时间、测试人员、测试时间
 - ii. 正文: 功能点、测试点、测试输入、预期结果、实际结果
 - iii. 用例执行结果统计
- d) 功能点模块化理念

都知道一个复杂庞大的系统, 程序在实现时会将其分成若干模块按照模块功能优先级进行实现。我们测试过程中也采用这种方法, 将复杂的功能点按照实现功能进行分类, 分类后的测试点, 再进行分类, 直至细分成为一条条用例。就像庖丁解牛那样。

按照等价类划分法, 将同一判断条件的测试点组成一个集, 在这个条件基础上再次判断的条件, 我们假设它已经成立。这样在用例设计过程中就需要测试人员清楚的知道, 哪些条件是一类需优先确认的, 哪些是以这类条件为基础的。我们最终形成的测试用例一定确保的是一条用例只检查一个测试点。

这样设计也有另外一个好处, 如果一条用例不能走通, 其它的还可以继续检测, 经常会遇到测试过程中由于一个 bug, 导致测试工作停滞。现在这样子我们就可以采取脚本调试, 或者其它方法跳过有 bug 的测试内容, 继续进行其它测试点的测试了。

- e) 场景测试法协助功能点细分

游戏测试中, 场景测试方法是经常用到的一种方法, 什么是场景测试法, 及按照功能设计要求, 在脑中模拟出来的一个功能使用时的操作流程。按照每步操作的针对点, 将针对点划分为所用例设计时的小功能点。划分时需每步针对点的各种检查点分到该功能点内设计为该功能点的检查点。再根据检查点进行测试输入(及操作过程)的编写。用例编写过程中的思考方式就如上了。讲起来比较抽象, 希望对

大家有所帮助。

- f) 用例的设计原则——一直有人问到底要详细到什么程度
 - i. 我们不期待用例编写到任何人都可以执行, 也没有这个必要
 - ii. 我们针对的是网游的测试人员, 至少是玩过网游的人, 这些人对于游戏中的基础设定都有认识, 我们不可能对着一个不知道任务界面是什么的人大讲怎么测试任务。所以我们用例编写的原则就是针对我们测试组内的测试人员。
 - iii. 但是, 请不要简略到别的测试人员看不懂, 特别是当你是专职的用例编写人员时, 编写时请多考虑下语言描述的方式。请让你的同伴可以看懂, 你所要表达的意思。
 - iv. 用例是没有固定格式的, 它的主要原则就是, 测试中所需所有信息, 我通过你的文档都能够获取到。所以不要再执着的像别人要模板。模板你自己都可以设计, 发挥你的创意。

四、编写过程注意事项

与设计人员的沟通

拿到一份文档时请不要急于编写, 在这之前很多事情需要做, 请先将文档阅读至少三遍, 然后思考下, 你自己大脑中是否有你所看文档功能点的一个流程图, 当确认已经准备好了。开始设计用例, 用例设计的过程就是与设计人员不断沟通, 深入了解功能的过程。你会发现, 或许跟你之前流程图中想像的并不完全一样。这个时候不必惊讶, 去找他们核对就好。不怕发现问题, 就怕没有发现问题, 最终做了很多无用功。编写过程中发现的没有预期结果的内容, 请及时与策划人员、程序人员核对, 必须三方核对。核对完毕提醒策划人员及时更新设计案, 提醒程序人员设计案新修改内容。这样你会发现, 设计测试用例过程的本身就是发现策划案不完善的过程。

请运用你的思维, 采用边界法、等价类划分法、错误推断法、以及以往的经验, 将每一个测试点的所有需检查点进行充分的设计。发挥你的主动性, 和测试组内其它人探讨你认为可能存在风险的测试点, 以便得到更多有价值的信息。Over