缺陷管理

## 软件缺陷的基本概念

主要分为：缺陷、故障失效。

缺陷：存在于软件之中的偏差，可被激活，以静态形式存在与软件内部。

故障：软件运行中出现的状态，可引起意外情况，如不加处理，可产生失效，是一个动态行为。

失效：软件运行时产生的外部异常行为结果，表现与用户需求不一致，功能能力终止，用户无法完成所需要的应用。

## 缺陷管理的目的

1. 缺陷跟踪

保证缺陷得到有效的跟踪和解决。

2.缺陷分析

获取正确的BUG信息，用作缺陷分析和产品度量。我们通过对BUG数据的分析，我们可以找到，为什么会产生这些BUG？找到原因，那么在后续的项目中我们就可以规避它。

产品度量：软件公司负责人最关心的就是产品的一些度量数据，这些数据也就决定了，我们的产品在那个时间点可以发布可以上市。

## 软件缺陷的相关属性

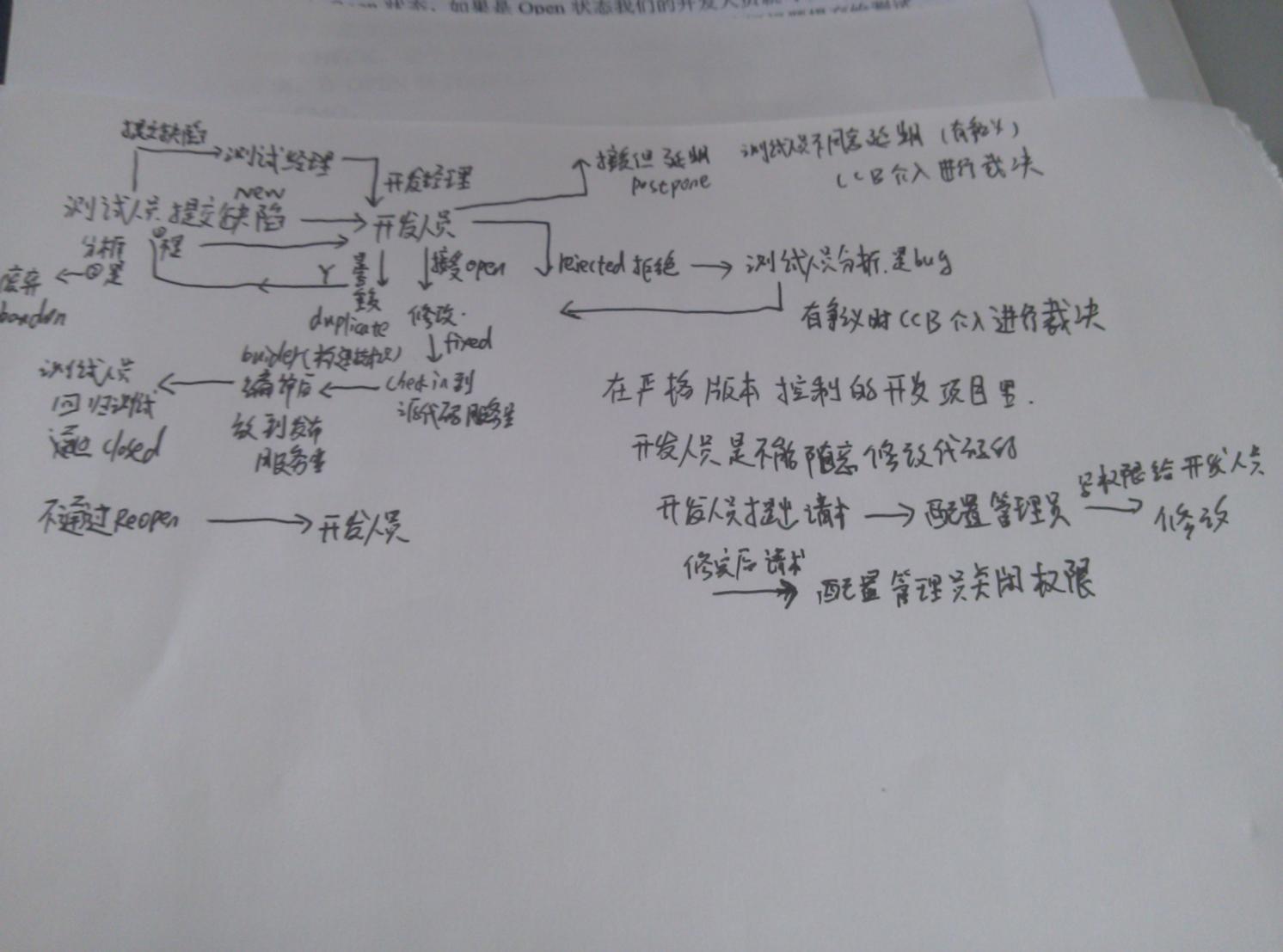
1. 缺陷发现人 2.缺陷发现时间3.缺陷所属版本4.缺陷修改日期5.软件缺陷的状态6.缺陷的严重程度。

|  |  |
| --- | --- |
| **New** | **缺陷的出初始状态** |
| **Open** | **开发人员开始修改缺陷** |
| **Fixed** | **开发人员修改缺陷完毕** |
| **Closed** | **回归测试通过** |
| **Reopen** | **回归测试失败** |
| **Postpone** | **推迟修改** |
| **Rejected** | **开发人员认为不是程序问题，拒绝缺陷** |
| **Duplicate** | **与已经提交的Defect重复** |
| **Abandon** | **被Rejected和Duplicate的缺陷，测试人员确定后认为的确不是问题，将Defect置为此状态** |

## 缺陷的严重程度

|  |  |
| --- | --- |
| 致命 | 软件的意外推出甚至操作系统崩溃，造成数据丢失。 |
| 严重 | 单个功能失效导致多个相关功能均失效。 |
| 一般 | 单个功能失效，不会影响其他功能 |
| 提示 | 软件界面的细微缺陷，例如，某个控件没有对齐，某个标点符号丢失。一般提示问题，不会影响整个软件的使用 |

## 软件缺陷的管理流程



## 软件缺陷跟踪单

|  |  |
| --- | --- |
| 缺陷项目 | 注意事项 |
| 简单描述 | 用一句简单的、提纲挈领地描述清楚问题 |
| 详细描述 | 1. 描述问题的基本环境，包括操作系统、硬件环境、网络环境、被测试软件运行的环境 2. 操作步骤 3. 如果从gui界面上可以反映出软件的异常，采用拷屏的方式截取界面，粘贴在附件中 4. 被测试软件运行时候的相关日志文件 5. 问题的简单分析（可选项） 6. 被测试软件的版本 7. 状态、严重级别、优先级别 8. 提交日期 提交人 |
| 相关附件 | 1.gui的拷屏图片2.被测试软件运行的相关日志文件 |