**测试管理功能需求整理**

2013-4-25

目录

[一、 背景及目的 2](#_Toc354671955)

[1.1、背景 2](#_Toc354671956)

[1.2、功能概要 2](#_Toc354671957)

[二、 功能分解 3](#_Toc354671958)

[2.1、 项目和模块管理 3](#_Toc354671959)

[功能描述 3](#_Toc354671960)

[相关表单 3](#_Toc354671961)

[流程图 4](#_Toc354671962)

[特殊要求 4](#_Toc354671963)

[2.2、测试需求管理 4](#_Toc354671964)

[功能描述 4](#_Toc354671965)

[相关表单 5](#_Toc354671966)

[2.3、测试版本管理 6](#_Toc354671967)

[功能描述 6](#_Toc354671968)

[相关表单 6](#_Toc354671969)

[2.4、Bug管理和处理 7](#_Toc354671970)

[功能描述 7](#_Toc354671971)

[相关表单 7](#_Toc354671972)

[流程图 8](#_Toc354671973)

[特殊要求 9](#_Toc354671974)

[2.5、测试用例管理 9](#_Toc354671975)

[功能描述 9](#_Toc354671976)

[2.6、组织管理 10](#_Toc354671977)

[功能描述 10](#_Toc354671978)

[性能要求 10](#_Toc354671979)

[相关表单 10](#_Toc354671980)

[2.6、测试数据分析 10](#_Toc354671981)

[功能描述 10](#_Toc354671982)

# 背景及目的

## 1.1、背景

测试管理工具中Bug管理是测试工作中必不可少的工具。尝试过Word，Excel，Groove，Bugzilla，zentao\_BugFree（V1.0），其中前三款并不属于专业的测试软件，测试的时候需要配置的信息比较多；Bugzilla免费版的安装部署复杂；zentao\_BugFree（V1.0）可以解决核心的功能需求，但是只是对Bug进行管理，不能和需求、用例、角色权限进行关联，随着项目的发展，越来感觉功能过于简陋，不能满足越来越多的使用想法。目前正在使用的为开源zentao\_PMS（V4.0）基本可以解决测试相关的需求，但是自主可控性不太好，同时从部门整体来讲工作平台不够集中。

为了促进内部工作协作、高效的进行，并将日常工作需要的办公软件集成到统一的平台上，特提出此研发需求。该功能的主要使用对象可分为三种：测试人员，研发人员，高层管理，各角色关注的信息重点不同。

希望此研发目的能达到解决：测试流程中的关键环节集中到URM5上。以下内容为可以想到的一些需求内容，仅供参考。

## 1.2、功能概要

该研发功能可分解为7个一级功能：项目和模块管理，测试需求管理，测试版本管理，Bug管理和处理，测试用例管理，组织管理，测试数据的分析（还有测试文档管理，可以推迟考虑），其中核心功能为：项目和模块管理，Bug管理和处理。

其次不同用户角色的关注的功能点不同：

* 测试人员角色需要使用的功能覆盖全部功能，但核心功能是提Bug，发布，及状态的标识。
* 研发人员角色的主要关注功能是查看已经发布的Bug，进行解决等。
* 高层管理角色的主要关注项目测试数据的统计、分析等。

以下为各部分的较详细功能描述。

# 功能分解

## 2.1、 项目和模块管理

### 功能描述

该功能要能完成测试项目的维护，项目中每块的维护目的。

1. 测试项目及项目模块的维护：项目添加、编辑、删除项，项目模块的添加、修改、删除。
2. 项目及模块间要以关联的关系展现，如类似组织架构的形式等。

### 相关表单

项目表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| ProjectID |  |  | 项目ID |
| ProName |  |  | 项目名称 |
| ProCreatedBy |  |  | 有谁创建 |

项目模块表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| ModuleID |  |  | 模块ID |
| ModName |  |  | 模块名称 |
| Parent |  |  | 父级模块 |
| ProjectID |  |  | 项目ID |

### 流程图

### 特殊要求

1. 可以对项目及项目的模块进行排序、移动。
2. 删除项目及项目模块时如果其下有Bug不能删除Bug。

## 2.2、测试需求管理

### 功能描述

该功能要能完成测试需求的登记、统计。

1. 需求维护：可以完成测试需求的新建、编辑、删除（新建时要能和所属的项目和模块关联）。
2. 需求的查看：可以按项目查看需求，可以按时间查看需求，可以按需求类别查看需求，可以按状态查看需求。
3. 统计分析：可以按项目、时间段对需求的类别进行图标统计分析。
4. 需求类别的维护：可以对需求类别进行新建、编辑、删除（如需求类别可以分为：内部，测试Bug，客户等）。
5. 需求状态管理：可以将需求设置为“激活”和“关闭”两种状态标志，并可以按这两种状态进行查看。

### 相关表单

需求表：Requirement Table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| RequirementID |  |  | 需求ID |
| ReqName |  |  | 需求名称 |
| ReqContent |  |  | 需求内容 |
| ReqState |  |  | 需求状态 |
| ReqCreatedBy |  |  | 有谁创建 |
| ReqClassID |  |  | 需求ID |
| ProjectID |  |  | 所属的项目ID |
| ModuleID |  |  | 所属的模块ID |

需求类别表：RequirementClass Table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| ReqClassID |  |  | 需求ID |
| ReqClassName |  |  | 需求名称 |

## 2.3、测试版本管理

### 功能描述

该功能要能完成测试版本的维护，可以完成测试版本的创建，产生本次测试的一个版本号，并且在创建测试版本时要能和相关的需求、Bug进行关联。

1. 版本维护：可以进行版本的新建、删除、修改（新建版本时包括版本名称，版本号，版本内容简介， 该版本相关联的需求或者Bug）。
2. 点击关联的需求或者Bug可以进行相关的需求或者Bug.
3. 可以对版本进行开始测试的操作，点击开始测试后，创建的版本后自动显示在新建Bug时的版本号列表中并且按创建的先后倒排序。

### 相关表单

测试版本表：Version Table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| VersionID |  |  | 版本ID |
| VerName |  |  | 版本名称 |
| VerNum |  |  | 版本号 |
| VerContent |  |  | 版本简介 |
| RelatedReqID |  |  | 关联的需求ID |
| RelatedBugID |  |  | 关联的BugID |

## 2.4、Bug管理和处理

### 功能描述

要能完成测试人员提交相应项目模块的Bug，批量发布，研发和测试的处理，并能多角度进行查询等。

1. 测试人员可以进行Bug的维护：可以在相应的项目模块下新建一个Bug，可以进行删除、修改。
2. 附件：能添加、删除、对附件重命名，能方便的查看附件内容。
3. BUG查看：

3.1点击项目、模块可以直接查看到该项目或者模块下的BUG；

3.2可以按照Bug信息中包含的任何字段进行准确检索、模糊检索等

1. Bug的发布：测试人员可以单个、批量选择BUG进行发布/可以单个、批量取消发布，研发人员只能看到发布的BUG。
2. 开发解决：开发人员可以查看到所有已经发布的、分配给自己的，可以进行解决，可以单个或批量选择BUG进行分配、解决。

### 相关表单

Bug表：Bug Table

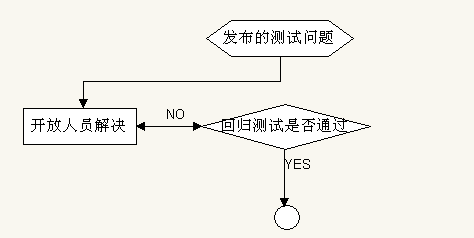
|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| BugID |  |  | BugID |
| BugName |  |  | Bug标题 |
| VerNum |  |  | 版本号 |
| BugContent |  |  | Bug详细内容 |
| BugMoudle |  |  | 所属的项目、模块 |
| BugBuilder |  |  | Bug创建人 |
| BugCreateDtate |  |  | 创建时间 |
| BugReceiver |  |  | 分配给谁 |
| BugGrade |  |  | Bug等级 |
| BugSolution |  |  | Bug解决方案 |
| BugSolutedDate |  |  | 解决时间 |
| BugState |  |  | Bug状态 |
| CloseDate |  |  | 关闭时间 |

说明：

1. Bug等级可以确定为：1,2,3,4
2. Bug状态可以确定为：激活，关闭，（默认状态可以为空，发布后的状态可以为激活）
3. Bug解决方案可以确定为：已修复，设计如此，重复提交，不修复，外部原因

### 流程图

简单的流程图见下（省略了问题的评审等环节）：



### 特殊要求

1. 每个模块后显示该模块下的Bug数
2. 选择一个模块点击建Bug，直接将该Bug创建在相应的模块下
3. 单击模块时显示该模块下的Bug列表

## 2.5、测试用例管理

### 功能描述

该部分功能要能对测试用例进行管理，包括新建、编辑、删除、不同用例间的关联等。

1. 测试用例的单个创建及批量创建，执行结果可以暂时确定为三种：N/A，通过，失败，失败的用例可以自动生成Bug；
2. 不同用例可以进行关联；

## 2.6、组织管理

### 功能描述

该部分功能要能完成对测试分组的维护和功能的权限控制。

1. 分组维护：可以新建一个分组，并能为该分组添加人员，能为该分组设置可以访问的项目权限；可以进行分组的编辑、删除；

### 性能要求

### 相关表单

分组表：Group Table

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 类型（长度） | 约束 | 说明 |
| GroupID |  |  | 分组ID |
| GroupName |  |  | 分组名称 |

## 2.6、测试数据分析

### 功能描述

该部分功能要能完成可以按项目分析测试数据。

1. 项目Bug走势图。
2. 项目或者各模块Bug严重程度分布等
3. 每天新增Bug数