报修系统设计文档

# 1 用户角色

## 1.1 报修用户

报修用户小明，学生，在举办社团活动的时候，发现三教某教室的投影仪麦克风没有声音。他登录学校报修系统，提交报修单。



## 1.2 维修部门主管

小明报修之后，维修部门主管收到一封电子邮件，上面有报修的信息。

主管登录报修系统，把这个报修单审核通过，然后这个报修单就显示在了“报修查询”页面。



同时，维修部门主管把这个报修分配给维修人员小王，小王收到一封电子邮件。

## 1.3 维修人员

维修人员小王收到电子邮件之后，上面写着故障详细情况。小王按照描述去三教修理。



修理完成后，小王填写故障处理结果，把报修单的状态变成“已处理”。



此时维修部门主管和报修人小明都收到一封电子邮件，提示已经处理完毕。

# 2 事件流

# 3 架构模式

使用MVC架构模式。

因为web程序使用MVC架构模式开发，可以使数据、逻辑和视图分开，方便维护和快速开发。

* （控制器 Controller）- 负责转发请求，对请求进行处理。
* （视图 View） - 界面设计人员进行图形界面设计。
* （模型 Model） - 程序员编写程序应有的功能（实现算法等等）、数据库专家进行数据管理和数据库设计(可以实现具体的功能)。

# 4 测试

下面列出了各个功能点的测试。

## 4.1 用户提交表单

数据库预先初始化一个普通用户、一个部门主管、一个维修人员。

模拟提交正常的维修表单，看是否能返回成功，且部门主管是否收到提醒邮件。

模拟提交非法的维修表单（一些必填项为空），看是否能返回失败。

## 4.2 部门主管分配维修任务

数据库预先初始化一个普通用户、一个部门主管、一个维修人员、一条未分配的维修任务、一条已分配的维修任务。

用curl模拟维修部门主管登录，模拟分配未分配的维修任务给一个维修人员，看是否能返回成功，且该维修人员是否收到提醒邮件。

用curl模拟维修员用户登录，模拟分配未分配的维修任务，看是否能返回失败（越权访问）。

用curl模拟维修部门主管登录，模拟分配已分配的维修任务给一个维修人员，看是否能返回失败。

## 4.3 维修人员完成任务

数据库预先初始化一个普通用户、一个部门主管、两个维修人员A和B、一条已分配给A的维修任务、一条A已完成的维修任务。

用curl模拟维修人员A登录，模拟完成一条分配给他的维修任务，看该任务状态是否变成已完成，报修人员、部门主管是否收到提醒邮件。

用curl模拟维修人员B登录，模拟完成一条分配给A的维修任务，看是否能返回失败。

用curl模拟维修部门主管登录，模拟完成一条分配给A的维修任务，看该任务状态是否变成已完成，报修人员、部门主管是否收到提醒邮件。（主管也可以完成任务）

用curl模拟维修人员A登录，模拟完成一条A已完成的维修任务，看是否能返回失败。

# 5 开发进度与Sprint分配

## Sprint 1

数据库的模型。

用户角色：普通用户、部门主管、维修人员。

用户登录系统。

报修单的数据模型。包含的数据库表字段。

报修单的状态：

* 未分配
* 已分配
* 已完成

## Sprint 2

用户提交报修单。

部门主管或维修人员查看报修单。

部门主管分配报修单给维修人员。

维修人员完成报修单。

## Sprint 3

用户查看自己提交的报修单的列表。

部门主管查看所有报修单的列表。

维修人员查看分配给自己的报修单列表。

## Sprint 4

提醒系统。

用户提交报修单后，给部门主管发送邮件提醒。

部门主管分配报修单后，给维修人员发送邮件提醒。

维修人员完成报修单后，给用户和部门主管发送邮件提醒。