#### 管理员账户的增删模块详细设计

#### 子模块1：添加管理员模块

##### 设计图

本模块的处理过程如图所示：

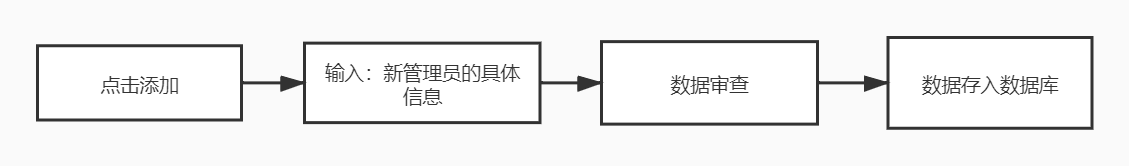


图1 添加管理员模块流程图

##### 功能描述

本模块通过触发管理员的请求，获取新管理员的具体账户，密码，邮箱，并对数据进行相应的审查，通过审查后将数据存入数据库并更新，生成新的管理员。将通过管理员的响应进行包装，将包装的响应转化为具体页面的响应。

##### 输入数据

添加管理员模块需要五项输入数据：用户名，密码，确认密码，手机，邮箱；

用户名是管理员的唯一标识，为一串字符串，长度为1-100，用于识别管理员。

密码是长度为5-20的字符串，用于核实用户账户信息。

确认密码为长度5-20的字符串，在与密码不同的情况下禁止添加管理员的申请，保证了密码的准确性。

手机号为符合中国手机号标准的11位数字组成，是管理员基础信息的一部分，通过此可以联系管理员。

邮箱为符合邮箱标准的字符串组成，是管理员基础信息的一部分，通过此可以联系和唯一识别管理员。

##### 输出数据

管理员增添模块输出一个前台包装管理员的JsonManager的包装数据，将包装数据转换为具体的管理员类，并存入数据库中。并且将请求的响应包装，返回至前端转化为响应的页面。

##### 业务算法和流程

从程序模块根据输入的管理员信息进行包装，对包装类进行审查，对数据库中的数据进行匹配，拒绝不符合标准的数据的请求。符合标准的请求转换为具体的管理员类，并存入数据库中，拒绝不符合标准的请求。将请求的响应包装，返回至前端转化为响应的页面。

##### 数据设计

将添加的管理员信息封装为JsonManager，通过前端传往后端；

private String info; 封装结果

将请求返回的网页封装为 JsonResponse；

private int status; 页面状态  
private String msg; 页面信息

将封装后的JsonManager 转换为Manager类,是管理员信息的封装类：

private Integer mId; 用户id

private String mName; 用户姓名

private String mPassword; 密码

private String mPhone; *手机号*

private String mEmail; *邮箱*

private Integer mStatus; *状态*

##### 源程序文件说明

本程序位于根目录下的ManagerController.java文件，是管理员控制模块主程序。

##### 函数说明

本程序包含的函数如下：

1. public String managerManage(@PathVariable Integer pageNo, @PathVariable Integer pageSz, HttpServletRequest request)：

模块的入口，输入页数,页面尺寸,Http请求，生成一个String字符串与相应的页面。

2） public JsonResponse addManager(JsonManager jsonManager)

模块的功能实现函数，通过将JsonManager的包装类进行转换后，判断是否添加成功，添加成功后将返回网页请求的JsonResponse的包装类。

3）public Manager parse()：

将JsonManager包装类转化为Manager类的函数

本程序包含的函数类如下：

1）Manager类：封装管理员信息的类。，

其中包含的函数如下：

public Integer getmId() 获取管理员id

public void setmId(Integer mId) 设置管理员id

public String getmName() 获取管理员名称

public void setmName(String mName) 设置管理员名称

public String getmPassword() 获取管理员密码

public void setmPassword(String mPassword) 设置管理员密码

public String getmPhone() 获取管理员手机号

public String getmEmail() 获取管理员邮箱

public void setmEmail(String mEmail) 设置管理员邮箱

public Integer getMStatus() 获取管理员状态

public void setMStatus(Integer status) 设置管理员状态

public String toString() 获取所有信息

2）JsonManager类：前端传递的封装管理员信息的类：

public Manager parse() 转换为Manager

public Map<String, String> toMap() 对信息进行加密

String getInfo() 获取封装结果

public void setInfo(String info) 设置封装结果

3）JsonResponse类：将请求返回的网页的封装类：

将请求返回的网页封装为 JsonResponse

public int getStatus() 获取网页状态

public void setStatus(int status)设置网页状态

public String getMsg()获取网页信息

public void setMsg(String msg)设置网页信息

#### 子模块2：删除管理员模块

##### 设计图

本模块的处理过程如图所示：

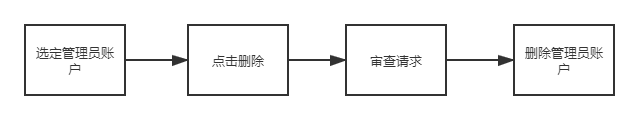


图1 删除管理员模块流程图

##### 功能描述

本模块通过管理员的请求，审查删除管理员的请求，删除管理员。

##### 输入数据

输入：管理员用户名，被删除管理员用户名。通过识别管理员唯一标准id，确认用户的管理员身份。并且通过被删除管理员用户名，确定被删除管理员的身份。

##### 输出数据

将具体的管理员数据删除后，管理员删除模块输出一个前台请求的JsonResponse包装数据，将包装数据转换为具体的页面响应。

##### 业务算法和流程

从程序模块根据输入的管理员信息进行包装，对包装类进行审查，对数据库中的数据进行匹配，拒绝不符合标准的数据的请求。符合标准的请求转换为具体的删除操作，将管理员数据从数据库中删除。将删除结果进行包装后返回，将返回结果转换为具体的页面响应。

##### 数据设计

将添加的管理员信息封装为JsonManager，通过前端传往后端；

private String info; 封装结果

管理员列表 ids为List<Integer>属性

Http请求request为HttpServletRequest属性

将请求返回的网页封装为 JsonResponse；

private int status; 页面状态  
private String msg; 页面信息

将封装后的JsonManager 转换为Manager类,是管理员信息的封装类：

private Integer mId; 用户id

private String mName; 用户姓名

private String mPassword; 密码

private String mPhone; *手机号*

private String mEmail; *邮箱*

private Integer mStatus; *状态*

##### 源程序文件说明

本程序位于根目录下的ManagerController.java文件，是管理员控制模块主程序。

##### 函数说明

本程序包含的函数如下：

1. public String managerManage(@PathVariable Integer pageNo, @PathVariable Integer pageSz, HttpServletRequest request)：

模块的入口，输入页数,页面尺寸,Http请求，生成一个String字符串与相应的页面。

2）public JsonResponse delManager(@RequestBody List<Integer> ids, HttpServletRequest request)

模块的功能实现函数，通过解析HttpServletRequest的请求获取管理员id，删除ids中相同id的管理员，判断是否删除成功，，返回网页请求的JsonResponse的包装类。

3）public Manager parse()：

将JsonManager包装类转化为Manager类的函数

4）public boolean remove()

删除列表中的元素

5）public Integer getmId()

获取管理员id

本程序包含的函数类如下：

1）Manager类：封装管理员信息的类。，

其中包含的函数如下：

public Integer getmId() 获取管理员id

public void setmId(Integer mId) 设置管理员id

public String getmName() 获取管理员名称

public void setmName(String mName) 设置管理员名称

public String getmPassword() 获取管理员密码

public void setmPassword(String mPassword) 设置管理员密码

public String getmPhone() 获取管理员手机号

public String getmEmail() 获取管理员邮箱

public void setmEmail(String mEmail) 设置管理员邮箱

public Integer getMStatus() 获取管理员状态

public void setMStatus(Integer status) 设置管理员状态

public String toString() 获取所有信息

2）JsonManager类：前端传递的封装管理员信息的类：

public Manager parse() 转换为Manager

public Map<String, String> toMap() 对信息进行加密

String getInfo() 获取封装结果

public void setInfo(String info) 设置封装结果

3）JsonResponse类：将请求返回的网页的封装类：

将请求返回的网页封装为 JsonResponse

public int getStatus() 获取网页状态

public void setStatus(int status)设置网页状态

public String getMsg()获取网页信息

public void setMsg(String msg)设置网页信息