

**沈阳科技学院**

**课程设计说明书**

**课程设计名称**  面向对象程序设计课程设计

**课程设计题目**

院 系 信息与控制工程系

专 业 计算机科学与技术

班 级 计算机2001

姓 名 颜奇昌

指导教师 高珊 职 称 无

2022年 10月

沈阳科技学院教务处

目录

[第一章简介 3](#_Toc28946)

[1.1功能简介 3](#_Toc10093)

[1.2技术简介 3](#_Toc3130)

[1.2.1 JAVA技术 3](#_Toc25319)

[1.2.2 Mysql数据库 4](#_Toc14746)

[第二章需求分析(分析哪些功能，画出功能模块图，每个功能介绍) 5](#_Toc10110)

[2.1用户登录流程 5](#_Toc16708)

[2.2页面管理 6](#_Toc23888)

[2.3 垃圾分别和管理 6](#_Toc20997)

[一、四大分类 6](#_Toc19078)

[2.4垃圾投放 8](#_Toc2450)

[第三章系统设计（类的设计，方法等） 9](#_Toc3442)

[3.1 数据库的设计 9](#_Toc20151)

[3.3.1数据表的建立 9](#_Toc13414)

[3.3.2 数据库和编译器（IDEA）的连接 11](#_Toc5713)

[3.2登录页面的设计 12](#_Toc18774)

[3.2.1 整体布局 12](#_Toc12518)

[3.3管理员权限的设计 14](#_Toc6027)

[第四章系统实现（主要功能模块的截图+少量代码+文字说明 14](#_Toc28171)

[4.1登录功能 15](#_Toc25586)

[4.2用户管理 16](#_Toc3070)

[4.3垃圾类别查询 17](#_Toc21699)

[4.4垃圾投放与回收 20](#_Toc23845)

[第五章测试 21](#_Toc27550)

[5.1测试的目的 21](#_Toc6724)

[5.2 测试方案设计 22](#_Toc19791)

[6.2.1 测试策略 22](#_Toc20396)

[6.2.2 测试分析 22](#_Toc11468)

[6.3 测试结果 23](#_Toc1209)

[第六章总结（技术上的总结） 23](#_Toc6969)

# 第一章简介

## 1.1功能简介

垃圾分类查询管理系统，对不懂的垃圾进行查询进行分类并可以预约上门回收垃圾。让用户自己分类垃圾，按国家标准自己分类，然后在网上提交订单，专门有人负责回收，统一回收到垃圾处理站，然后工人开始再次分类，将可再次循环用的贩卖给工厂（以后有钱自己开）。订单处理完（一般7天内），将一部分钱返还给用户。让垃圾变成钱！

## 1.2技术简介

### 1.2.1 JAVA技术

Java主要采用CORBA技术和安全模型，可以在互联网应用的数据保护。它还提供了对EJB（Enterprise JavaBeans）的全面支持，java servlet API，SSM（java server pages），和XML技术。

JAVA语言功能：

面向对象：面向对象是Java编程语言的标志之一，是一种软件开发方法。最重要的是将所有东西变成对象，然后以某种方式编程。编程时，代码和数据写在每个对象上。 面向对象编程方法的出现使得人们在编程过程中的设计思考和操作变得非常简单，同时也提高了程序的安全性。

跨平台：Java流行的一个关键特性是它的跨平台特性，这使得用Java编程变得容易。您可以用Java编写程序并在其他地方运行它，而无需在编译后更改它。

垃圾回收机制：用来将那些在程序不操作时无用的对象所占用的内存空间释放掉，C ++最被人厌恶的就是因为其不能将在编程的过程中所占用的内存空间进行及时的释放，导致随着编程时间的变长所占用的内存空间越来越多。对于一些编程高手而言，他们会在刚开始编程的时候配置一块内存地址放在堆栈上，然后在不需要的时候会对其进行释放，而一些新手和菜鸟在很多的时候会忘记删除这个内存地址，从而导致程序在运行的过程中会变得十分的不稳定，最终有可能会导致程序崩溃。所以很多C ++的高手在编写程序时往往都会将删除后的指针的值设置为NULL，然后在删除之前确定一个指针的值是否为NULL。

## 1.2.2 Mysql数据库

数据库是系统开发过程中不可或缺的一部分。 在WEB应用方面，MySQL AB开发了一个具有很大优势的MySQL关系数据库管理系统。 MySQL可以将数据存储在不同的表中，这非常灵活，并且还可以提高系统在实际应用中的速度。 数据库访问最常用于标准SQL语言，MySQL用于SQL语言，因此它具有高度兼容性。数据库的操作是必不可少的，包括对数据库表的增加、删除、修改、查询等功能。现如今，数据库可以分为关系型数据库和非关系型数据库，Mysql属于关系性数据库，Mysql数据库是一款小型的关系型数据库，它以其自身特点：体积小、速度快、成本低等，Mysql数据库是目前最受欢迎的开源数据库。

在WEB应用技术中， Mysql数据库支持不同的操作系统平台，虽然在不同平台下的安装和配置都不相同，但是差别也不是很大，Mysql在Windows平台下两种安装方式，二进制版和免安装版。安装完Mysql数据库之后，需要启动服务进程，相应的客户端就可以连接数据库，客户端可通过命令行或者图形界面工具登录数据库。

# 第二章需求分析(分析哪些功能，画出功能模块图，每个功能介绍)

## 2.1用户登录流程

对于系统的安全性的第一关，就是用户想要进入系统，必须通过登录窗口，输入自己的登录信息才可以进行登录，用户输入的信息准确无误后才可以进入到操作系统界面，进行功能模块的相对应操作，如果用户输入的信息不正确，则窗口出现提示框，用户登录失败，返回到第一步进行重新输入，如图2-1所示。



2-1 用户登录流程图

## 2.2页面管理

以实现管理员 ，用户管理 ,页面管理 ,角色管理,首页,贡献管理 ,垃圾管理,全国统计,搜索记录,分类管理,分类列表,垃圾列表根据系统功能需求建立的模块相关如下图：



2-2页面管理流程图

## 2.3 垃圾分别和管理

一、四大分类

目前中国生活垃圾一般可分为四大类：可回收垃圾、厨余垃圾、有害垃圾和其他垃圾。将其四大类进行分类并以适当的方法进行处理。

1、可回收物

定义：可回收物(再生资源)是指回收后经过再加工可以成为生产原料或者经过整理可以再利用的物品，主要包括废纸类、塑料类、玻璃类、金属类、电子废弃物类、织物类等。主要种类：

废纸类-——报纸、纸箱板、图书、杂志、各种本册、其它干净纸张、各类利乐包装牛奶袋、饮料盒(需冲洗晾干)。

塑料类-——各种塑料饮料瓶、塑料油桶、塑料盆(盒)。

玻璃类--玻璃瓶、平板玻璃、镜子。

金属类——铝质易拉罐，各类金属厨具、餐具、用具，其它民用金属制品。

电子废弃物类——各类家用电器产品。

织物类--桌布、衣服、书包等。

2.厨余垃圾

定义：狭义的厨余垃圾是有机垃圾的一种，分为熟厨余包括剩菜、剩饭、菜叶；生厨余垃圾包括果皮、蛋壳、茶渣、骨、贝壳；泛指家庭生活饮食中所需用的来源生料及成品(熟食)或残留物。但广义的厨余垃圾还包括用过的筷子，食品的包装材料等。

3、有害垃圾

定义：有毒有害垃圾是指存有对人体健康有害的重金属、有毒的物质或者对环境造成现实危害或者潜在危害的废弃物。

主要包括：废药品、废杀虫剂、废消毒剂、废油漆、废溶剂、废矿物油、废化妆品、废胶片、废相纸、废荧光灯管、废温度计、废血压计、废充电电池、废扣子电池、碱性电池、锂电池、镍镉电池等。

4.其他垃圾

定义：包括除上述几类垃圾之外难以回收的废弃物，通常根据垃圾特性采取焚烧或者填埋的方式处理。

主要包括：使用过的卫生纸、复写纸、传真纸、照片、离心纸、蜡纸、转印纸、塑料光面废纸、妇女用卫生巾、婴儿纸尿布、餐巾纸、烟蒂、陶瓷制品、衣服、鞋类、石棉瓦、白板、木质玩具、雨鞋、木质家具、橡胶制品、轮胎等。

具体流程图如图2-3所示：



2-3垃圾分别和管理

## 2.4垃圾投放

将垃圾分类进行投放到固定点，经由专门人员进行运送到固定地方并进行处理。

也可以在系统内，进行登录投放垃圾信息，以及一些个人信息。依次累计记录可以获得一些适当的奖励。流程图如图2-4所示。



图2-4 垃圾投放

# 第三章系统设计（类的设计，方法等）

# **3.2登录页面的设**计

### 3.2.1 整体布局

采用了html语句和css以及js等语言，构成了简单的登录页面。

部分代码如下：

HTMl代码：

<div class="box">

<div class="title">登录</div>

<div class="input">

<label for="name">用户名</label>

<input type="text" name="name" id="name">

<span class="spin"></span>

</div>

<div class="input">

<label for="pass">密码</label>

<input type="password" name="pass" id="pass">

<span class="spin"></span>

</div>

<div class="button login">

<button onclick="login()">

<span>登录</span>

<i class="fa fa-check"></i>

</button>

</div>

</div>

css代码：color\_1 {  
 color: #3F85F1;  
 padding-left: 2px;  
 }  
 .color\_2 {  
 color: #ec4335;  
 padding-left: 2px;  
 }  
 .color\_3 {  
 color: #FCBC27;  
 padding-left: 2px;  
 }  
 .color\_4 {  
 color: #2FA858;  
 padding-left: 2px;  
 }  
 .new\_label {  
 display: inline-block;  
 padding: 4px 6px;  
 margin: 2px 4px;  
 line-height: 16px;  
 vertical-align: baseline;  
 white-space: nowrap;  
 background-color: #ecf0f1;;  
 color: #404040;

## 3.3管理员权限的设计

管理员可以利用数据库对一些用户进行增删改查。

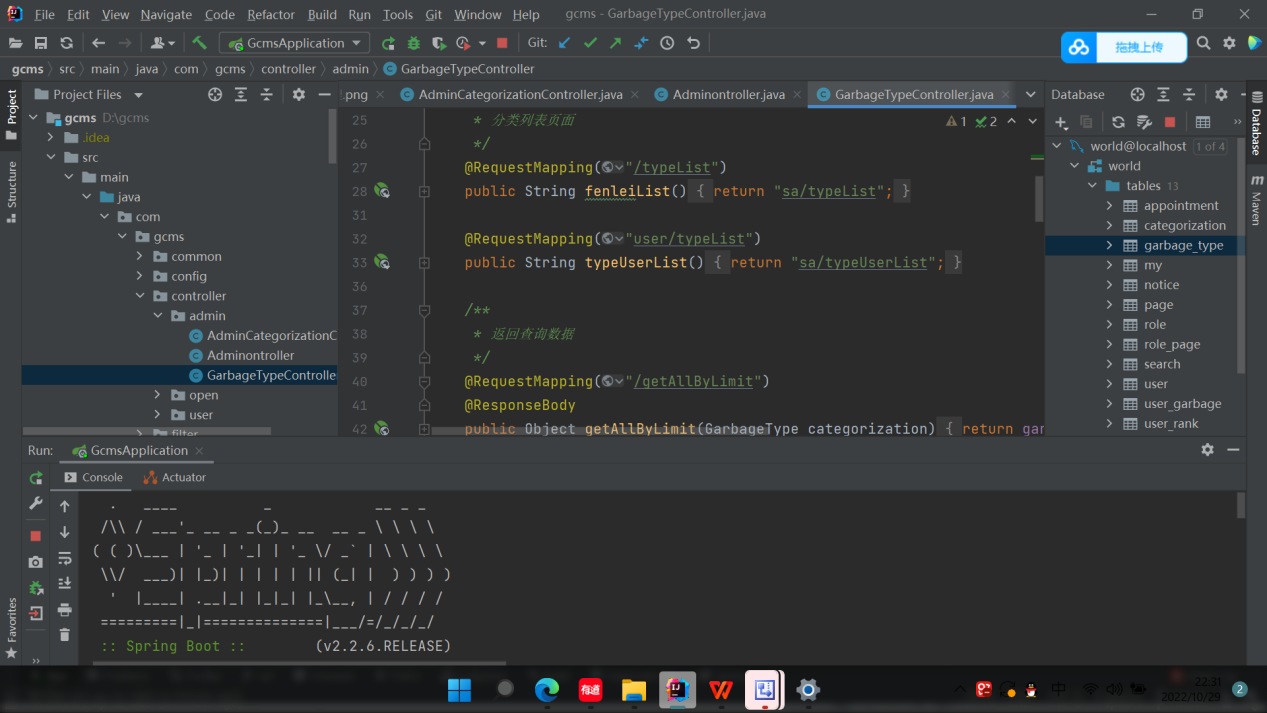
定义了String delRole，以次对用户的信息进行删除。如果用户信息删除成功则 return "ERROR"; return "SUCCESS";

删除角色\_权限表：boolean flag1 = roleService.delRoleById(id);

删除某个角色的所有用户 boolean flag2 = pageRoleService.delPageRoleByRoleId(id); boolean flag3 = userRoleService.delUserRoleByRoleId(id);

增添用户:定义了 addUser， user.setPassword， user.setCreateTime  
 userService.addUser等类进行用户增添。如成功则 return "SUCCESS".

# 第四章系统实现（主要功能模块的截图+少量代码+文字说明



## 3.1 数据库的设计

数据库是信息系统的基础和核心。数据库设计的好坏直接影响到信息系统开发的成败。创建数据库表首先确定实体的属性和实体之间的关系。根据关系创建一个数据表。

### 3.3.1数据表的建立

我们可以根据数据结构的详细分析要求，我们根据输入和输出数据量的要求进行分析，确定什么表表，结构之间的关系，我们可以验证，调整和完善，查询和浏览过程，可以实现数据库，以使用户对数据和功能有更多要求。

Appointment 表

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | user\_id | handle\_user\_id | phone | address | app\_time | info | create\_time | status | province | area | city |
| 1 | 1 | 1 |  |  | ######## | 哈哈哈 | ######## | 2 |  |  |  |

appointment表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | tip | Include | ask | type | create\_time |  |
| 1 | 废纸张、废塑料、废玻璃制品、废金属、废织物等适宜回收、可循环利用的生活废弃物 | 酱油瓶、玻璃瓶、平板玻璃、易拉罐、饮料瓶、洗发水瓶、塑料玩具、书本、报纸、广告单、纸板箱、衣服、床上用品等 | 轻投轻放  清洁干燥，避免污染，费纸尽量平整  立体包装物请清空内容物，清洁后压扁投放  有尖锐边角的、应包裹后投放 | 可回收垃圾 | ######## |  |
| 2 | 即其他垃圾，是指除可回收物、有害垃圾、湿垃圾以外的其它生活废弃物 | 餐盒、餐巾纸、湿纸巾、卫生间用纸、塑料袋、食品包装袋、污染严重的纸、烟蒂、纸尿裤、一次性杯子、大骨头、贝壳、花盆、陶瓷等 | 尽量沥干水分  难以辨识类别的生活垃圾投入干垃圾容器内 | 干垃圾 | ######## |  |
| 3 | 日常生活垃圾产生的容易腐烂的生物质废弃物 | 剩菜剩饭、瓜皮果核、花芬绿植、过期食品等 | 纯流质的食物垃圾、如牛奶等，应直接倒进下水口  有包装物的湿垃圾应将包装物去除后分类投放、包装物请投放到对应的可回收物或干垃圾容器 | 湿垃圾 | ######## |  |
| 4 | 废电池、废灯管、废药品、废油漆及其容器等对人体健康或者自然环境造成直接或者潜在危害的生活废弃物 | 废电池、油漆桶、荧光灯管、废药品及其包装物等 | 投放时请注意轻放  易破损的请连带包装或包裹后轻放  如易挥发，请密封后投放 | 有害垃圾 | ######## |  |

Role 表

|  |  |  |
| --- | --- | --- |
| role\_id | name | desc |
| 1 | 超级管理员 | 超级管理员 |
| 2 | 普通用户 | 普通用户 |

Categorization 表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | content | type | view\_count | value | harm | info |
| 1 | 啤酒瓶盖 | 1 | 19 | 5 | 0 | 啤酒瓶盖子 |
| 2 | 酒瓶盖 | 1 | 0 | 5 | 0 | 啤酒瓶盖子 |
| 3 | 瓶盖 | 1 | 0 | 5 | 0 | 饮料瓶盖子 |
| 4 | 酒瓶 | 1 | 0 | 5 | 0 | 啤酒瓶 |
| 5 | 酒瓶塞子 | 2 | 1 | 5 | 0 | 木质塞子 |
| 6 | 刀片 | 1 | 10 | 5 | 0 | 铁质刀片 |
| 7 | 钨丝 | 1 | 10 | 5 | 3 | 电灯钨丝 |
| 8 | 猫粮 | 3 | 10 | 3 | 1 | 小猫的粮食 |
| 9 | 塑料袋 | 2 | 10 | 1 | 4 | 日常用 |
| 10 | 塑料盒 | 1 | 10 | 5 | 0 | 日常用 |
| 11 | 废旧电池 | 4 | 22 | 1 | 4 | 有害垃圾 |
| 12 | 5号电池 | 4 | 18 | 1 | 4 | 有害垃圾 |
| 13 | 充电电池 | 4 | 13 | 1 | 4 | 有害垃圾 |
| 14 | 电池 | 4 | 13 | 1 | 4 | 有害垃圾 |
| 15 | 瑜伽垫 | 1 | 16 | 3 | 3 | 运动产品 |
| 16 | 打印过的废纸 | 1 | 14 | 4 | 1 | 纸质产品 |
| 17 | 巧克力 | 3 | 10 | 1 | 2 | 巧克力 |
| 18 | 荔枝 | 3 | 10 | 1 | 1 | 荔枝 |
| 19 | 荔枝壳 | 3 | 10 | 1 | 1 | 荔枝壳 |
| 20 | 荔枝核 | 3 | 10 | 1 | 1 | 荔枝核 |
| 21 | 废旧衣物 | 1 | 14 | 2 | 3 | 衣服 |
| 22 | 废旧衣服 | 1 | 10 | 2 | 3 | 衣服 |
| 23 | 笔记本电脑 | 1 | 22 | 2 | 3 | 笔记本电脑 |

### 3.3.2 数据库和编译器（IDEA）的连接

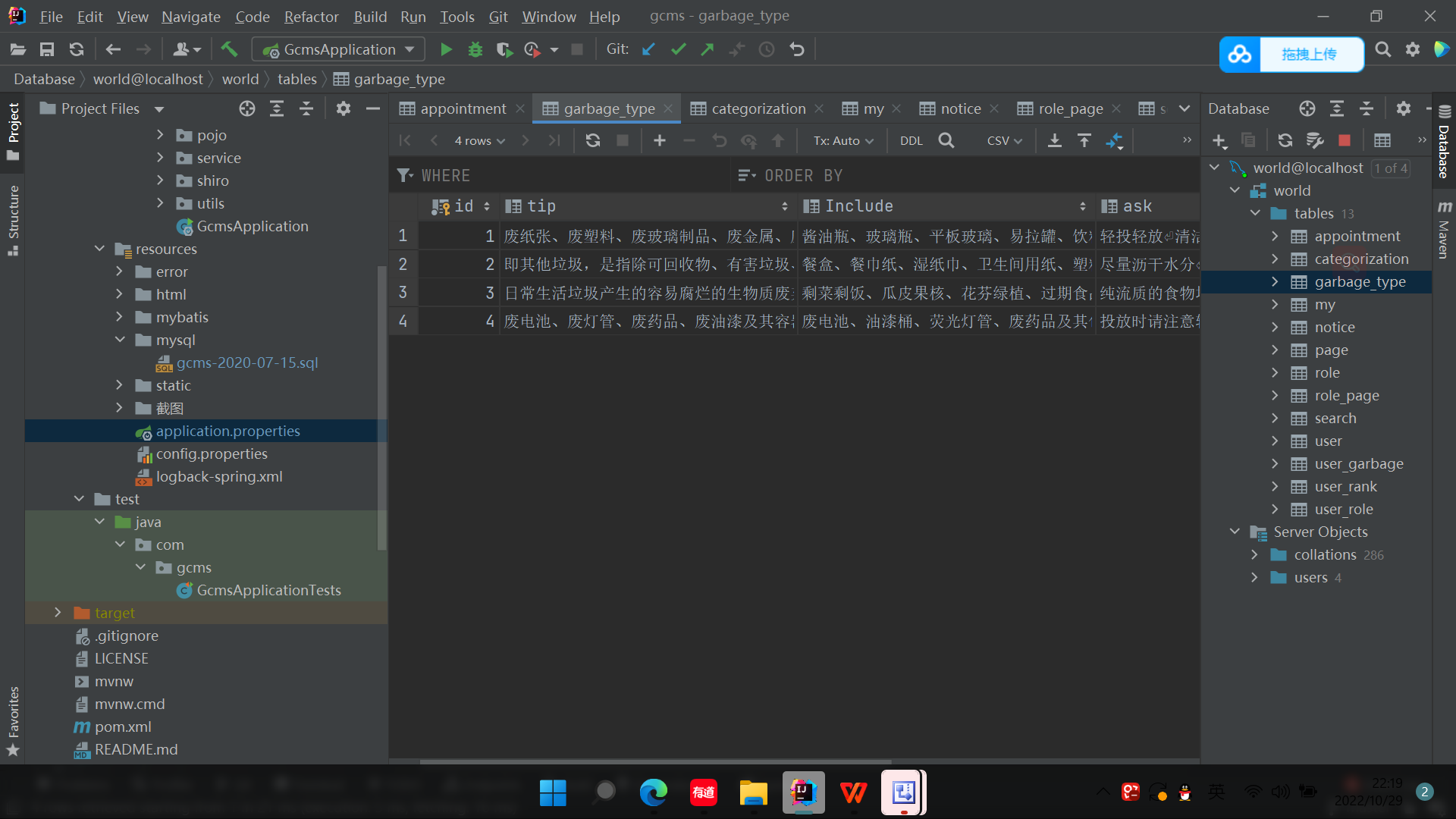
代码：

spring.datasource.url=jdbc:mysql://127.0.0.1:3306/YQC?characterEncoding=UTF-8&serverTimezone=GMT%2b8&useSSL=false

spring.datasource.username=root

spring.datasource.password=677520

截图：



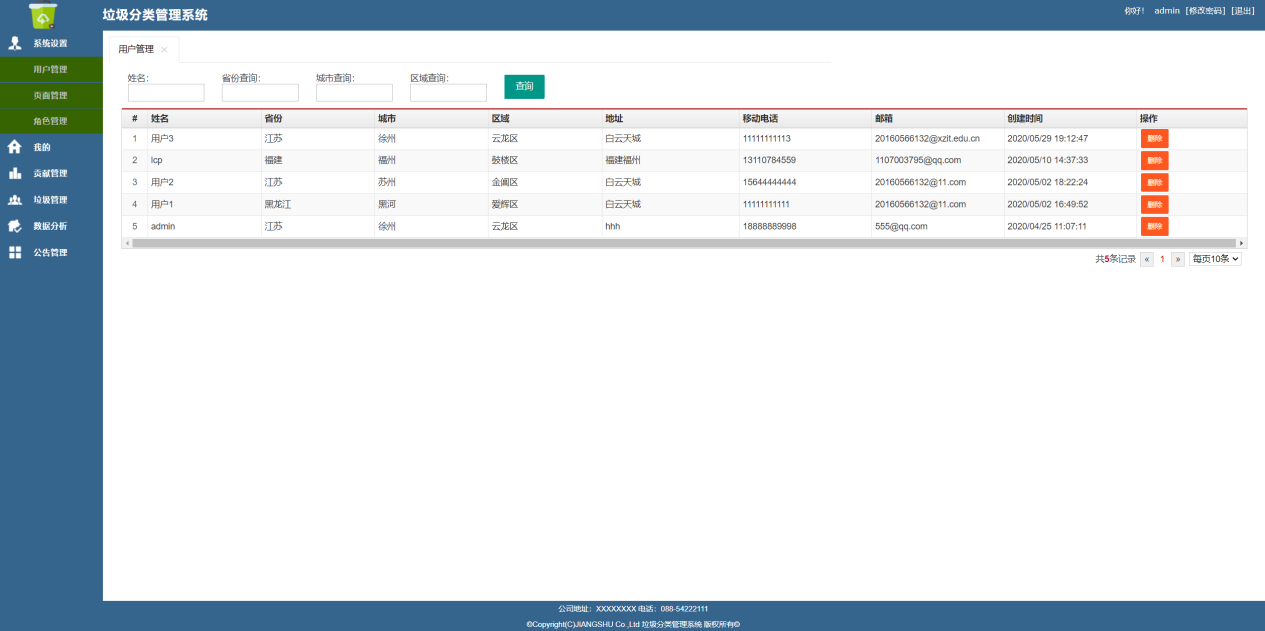
## 4.2用户管理

此功能为管理员进行操作，以次来管理普通用户信息。可以用来删除和记录用户的信息

public String delRole(int id) {  
 // 删除角色  
 boolean flag1 = roleService.delRoleById(id);  
 // 删除角色\_权限表  
 boolean flag2 = pageRoleService.delPageRoleByRoleId(id);  
 // 删除某个角色的所有用户  
 boolean flag3 = userRoleService.delUserRoleByRoleId(id);  
 if (flag1 && flag2 && flag3) {  
  return "SUCCESS";  
 }  
 return "ERROR";  
}

用户添加

public String addUser(User user) {  
 try {  
 user.setPassword(MD5.md5(user.getPassword()));  
 user.setCreateTime(new Date());  
 userService.addUser(user);  
 User u = userService.getUserByPhoneAndName(user.getPhone(), user.getName());  
 String[] ids = new String[1];  
 ids[0] = u.getId()+"";  
 userRoleService.addUserRole(3, ids);  
 C  
 } catch (Exception e) {  
 return "ERR";  
 }  
}



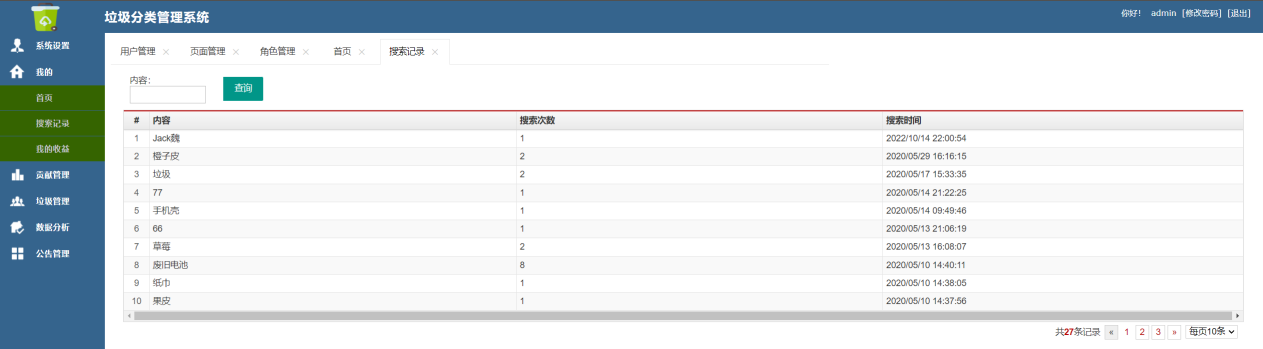
## 4.3垃圾类别查询

连接数据库，数据库中体现了一些垃圾的分类。用户可以通过搜索垃圾，依次了解自己垃圾的类别。

定义了Categorization类，实现垃圾的分类查询。Id类为垃圾名，以及一些其他类，比如：harm为垃圾的危害等级等。部分代码如下：

else if (main==2){  
 for (int k = 0; k >-1 ; k++) System.err.println("增加垃圾");  
System.out.println("-1增加垃圾处理处理方案-2查询全部处理方案-3修改处理信息-4返回");  
 System.out.println("请选择操作");  
 int solve = scanner.nextInt();  
 if (solve==1){  
 System.err.println("增加方案");  
 System.out.println("以上是全部垃圾查询报告")

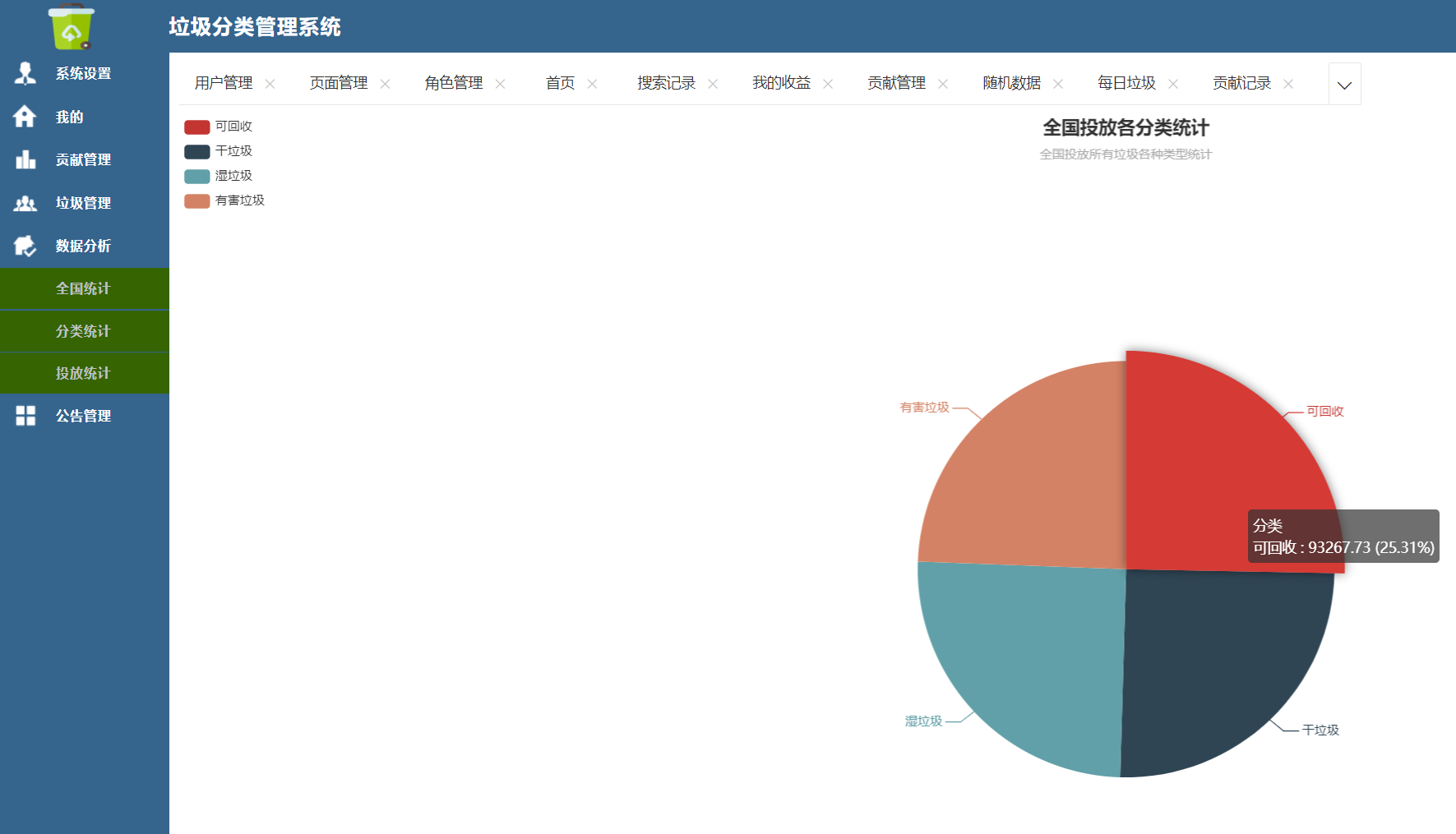
System.out.println("请选择您想针对的垃圾分类报告");  
 String id  
 continue;  
 }  
 else if (solve==2){  
 continue;  
 }  
 else if (solve==3){  
 continue;  
 }  
 else {  
 break;  
 }  
 }  
 }



# 第五章测试

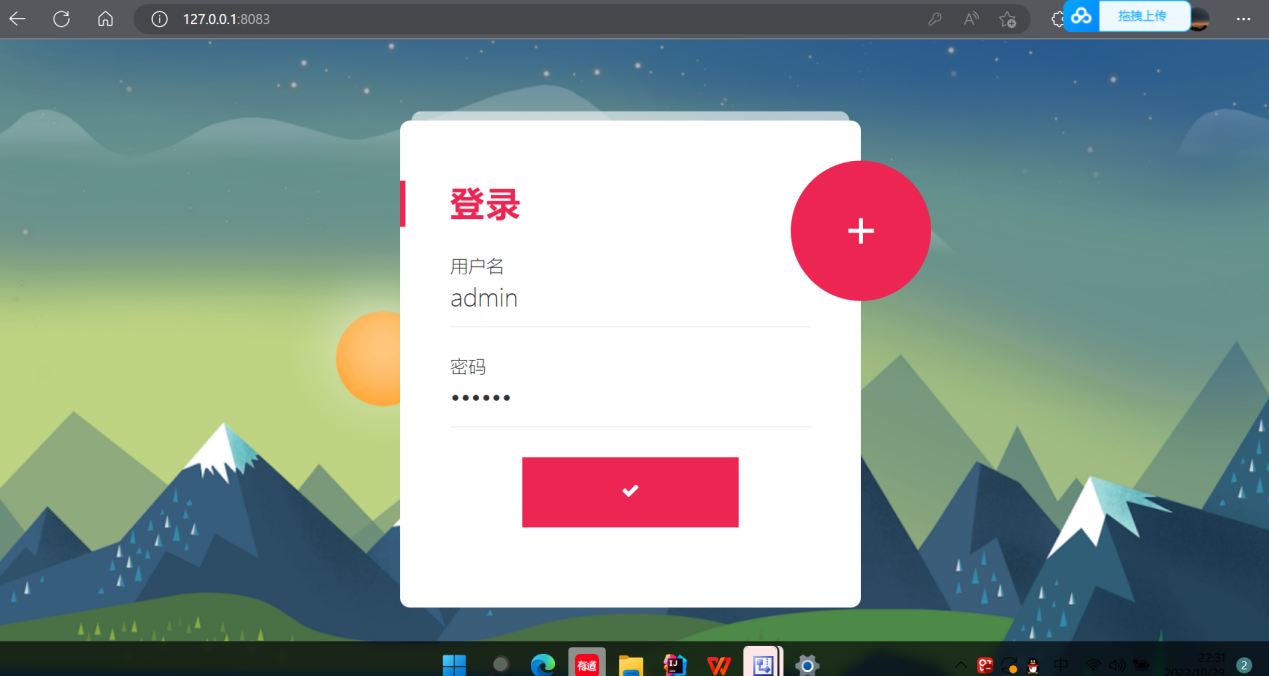
## 4.4垃圾投放与回收

用户可以查询自己的投放记录，以及全国的投放记录。



## 4.1登录功能

运行代码后，点击<http://127.0.0.1:8083/>网址进行登录以及后续的操作。



Map<String, String> filterChainDefinitionMap = new LinkedHashMap<>();

// 用户，需要角色权限 “user”

//filterChainDefinitionMap.put("/user/\*\*", "roleOrFilter[家长|老师|超级管理员]");

// 这样写必须同时拥有两个角色才能访问

// filterChainDefinitionMap.put("/user/\*\*", "roles[\"组员, 管理员\"]");

// 管理员，需要角色权限 “admin”

//filterChainDefinitionMap.put("/sa/\*\*", "roles[超级管理员]");

//filterChainDefinitionMap.put("/jz/\*\*", "roleOrFilter[家长|教师|超级管理员]");

//filterChainDefinitionMap.put("/ls/\*\*", "roleOrFilter[教师|超级管理员]");

// 开放登陆接口

filterChainDefinitionMap.put("/login", "anon");

filterChainDefinitionMap.put("/index", "anon");

filterChainDefinitionMap.put("/home", "anon");

filterChainDefinitionMap.put("/open/\*\*", "anon");

filterChainDefinitionMap.put("/upload/\*\*", "anon");

filterChainDefinitionMap.put("/file/\*\*", "anon");

filterChainDefinitionMap.put("/regist", "anon");

filterChainDefinitionMap.put("/doRegist", "anon");

// 静态资源设置为可访问

filterChainDefinitionMap.put("/css/\*\*", "anon");

filterChainDefinitionMap.put("/imgs/\*\*", "anon");

filterChainDefinitionMap.put("/js/\*\*", "anon");

filterChainDefinitionMap.put("/plug/\*\*", "anon");

filterChainDefinitionMap.put("/samples/\*\*", "anon");

## 5.1测试的目的

程序设计不能保证没有错误，这是一个开发过程，在错误或错误的过程中难以避免，这是不可避免的，但我们不能使这些错误始终存在于系统中，错误可能会造成无法估量的后果 如系统崩溃，安全信息，系统无法正常启动，导致安装用户手机屏幕等，为了避免这些问题，我们需要测试程序，并发现这些问题，并纠正它们 ，并使系统更长时间稳定成熟，本章的作用是发现这些问题，并对其进行修改，虽然耗时费力，但长期非常重要和必要 系统的开发。

软件测试与开发过程是一样的，都必须按照软件工程的正规原理进行，遵守管理学理论。不过，目前国内的软件测试已经积累了大量经验和方法，步骤相对成熟，软件测试的效率也越来越高。

垃圾分类系统的实现，对于系统中功能模块的实现及操作都必须通过测试进行来评判系统是否可以准确的实现。在垃圾分类系统 代码的运行前必须做的一步就是系统测试，对于测试发现的错误及时修改处理，保证系统准确无误的供给用户使用。

## 5.2 测试方案设计

### 6.2.1 测试策略

1、功能测试

从用户的角度来看，测试时不了解新开发软件的内部结构，因此可以将系统与黑匣子进行比较，盲目输入后可以查看系统给出的反馈。这种测试属于黑盒测试，在测试中如果输入错误信息系统会报错。

2、性能测试

测试程序的整体状况（称为性能测试）通常使用自动化测试工具来检测系统的整体功能，在负载测试和压力测试之间进行分配，在某些情况下，将这两种情况结合起来。虽然压力测试可以检测到系统可以提供的最高级别的服务，但负载测试可以测试系统如何响应增加的负载。

### 6.2.2 测试分析

测试评估系统质量的方法不局限于系统编码和过程，应该与软件设计工作和历史需求分析密切相关。

软件测试应遵循以下原则：

(1)软件测试应尽快进行，整个测试部分应在软件开发和设计的整个过程中进行。如发现错误，立即处理，将大大减少软件开发的时间，并提高软件的质量。

(2)在软件的各种测试中，测试过程中使用的计划、报告等应妥善处理和存储。其主要目的是为了便于以后系统的维护。

(3)软件测试整个过程中的聚类现象应优先考虑。

(4)对于软件测试，我们应该尽量不去和自己设计的系统进行参考，而是要测试对方的程序，以确保软件测试结果的客观性和公平性。

(5)整个测试计划严格按照软件测试的具体实施细则进行。

(6)对整个测试结果进行综合检查，尽量避免重复错误。

本垃圾分类系统满足用户的要求和需求，本垃圾分类系统小程序的使用能够有效的提高用户的使用率。

## 6.3 测试结果

测试后得到的性能和用例，系统具有足够的正确性、可靠性、稳定性，并且可以对输入数据进行准确的点击操作处理和响应测试用户的体验也得到了很好的反馈和响应时间。合理的范围，可以兼容所有主流浏览器，设计所需的效果。

# 第六章总结（技术上的总结）

在这个设计中，我花了大量的时间去理解系统开发中使用的知识，经过这段时间的努力工作最终完成了系统设计。通过这一阶段的学习，我发现了自己的不足，充分掌握了必要的应用技能，进一步的学习使我充实了自己的知识基础，完成了这个艰巨的任务。当遇到问题时，我很及时的寻求老师的帮助，通过专业的网站和论坛来解决，他们的帮助让我一步一步的成功克服了困难的问题。系统设计过程不容易，你需要不断充实自己，有勇气克服困难。系统开发的一些功能还不完善，需要继续改善后，通过用户体验来修改设计完美的系统，让用户得到更好的体验，我觉得很高兴，因为这是我第一次通过自己的努力实现这个系统，但绝不是我的最后一个，在未来我将努力实现更多的优秀的系统。

在一些编程语言的系统实现中，对词汇表不太熟悉，导致了开发的困难，但是我通过了合适的字典软件来解决这个大问题。由此，我学会了自己的英语缺陷。在那之后，我不断地提高自己的英语知识，这样我就不会有任何未来的工作和生活。毕业设计过程我感觉很深刻，从一开始就不熟悉开发技术，一步一步的使用，接触到文献和信息，不难理解，系统是一次又一次的实现，系统本身对于在线学习是有用的。我从这个设计中获益良多，论文的编写需要有自己的意愿去实现一点，学习生活中所有的问题的勇气，学习的过程就是学习的过程。毕业设计，我学会了将理论知识应用于实践。让我知道该怎么做，我们必须认真对待。勇于克服困难，相信未来，我会做得更好。