

本 科 生 毕 业 设 计（或论文）

论 文 题 目 基于Vue3+NodeJS的苗木交易平台的设计与实现

作 者 姓 名 李鑫

所学专业名称 软件工程

指 导 教 师 张燕玲

2023年4月3日

学 生： （签字）

学 号：

论文答辩日期：

指 导 教 师： （签字）

目录

[摘要 2](#_Toc3639)

[Abstract 2](#_Toc25350)

[1 绪论 3](#_Toc28530)

[1.1 设计背景及意义 3](#_Toc2512)

[1.2 设计研究目的 3](#_Toc6980)

[1.3 本文主要工作和内容 3](#_Toc9970)

[1.4论文的组织方法 3](#_Toc24251)

[2 系统开发环境介绍 4](#_Toc5657)

[2.1 Android平台基础知识 4](#_Toc17985)

[2.2 Android用户界面简介 4](#_Toc7088)

[2.3 开发工具介绍 5](#_Toc28569)

[2.3.1 Android Studio介绍 5](#_Toc9582)

[2.3.2 关键技术介绍 5](#_Toc9395)

[2.4本章小结 6](#_Toc25672)

[3 系统分析与设计 6](#_Toc20045)

[3.1需求分析 6](#_Toc13085)

[3.2 可行性分析 6](#_Toc1151)

[3.2.1 技术可行性 6](#_Toc6036)

[3.2.2 操作可行性 6](#_Toc29846)

[3.2.3 经济可行性 6](#_Toc22269)

[3.3 系统功能分析 6](#_Toc20212)

[3.4 系统数据库设计 7](#_Toc10042)

[3.4.1实体关系E-R图 7](#_Toc17618)

[3.4.2 数据表字段设计 9](#_Toc1022)

[3.5本章小结 10](#_Toc7436)

[4 系统实现 10](#_Toc13022)

[4.1系统开发环境 10](#_Toc24679)

[4.2 数据库连接配置 11](#_Toc7061)

[4.3 系统注册登录 13](#_Toc15713)

[4.4 资讯功能实现 14](#_Toc7158)

[4.5 预约功能实现 14](#_Toc18617)

[4.6用户管理 16](#_Toc27548)

[4.7评论功能实现 18](#_Toc16221)

[4.8 18](#_Toc10001)

[5 系统测试 18](#_Toc15997)

[5.1测试概述 18](#_Toc26186)

[5.2测本章小结试方案 18](#_Toc22437)

[5.3 测试用例 19](#_Toc28803)

[5.4 测试结果分析 21](#_Toc10634)

[5.4.1系统注册登录测试结果 21](#_Toc28828)

[5.4.2资讯管理测试图 21](#_Toc11317)

[5.4.3预约管理模块测试图 22](#_Toc21741)

[5.4.4用户管理模块测试结果 23](#_Toc24559)

[5.4.5评论管理模块测试结果 23](#_Toc10333)

[5.5本章小结 24](#_Toc26600)

[结束语 24](#_Toc7991)

[参考文献 24](#_Toc5674)

[致谢 26](#_Toc7839)

基于Vue3+NodeJS的苗木交易平台的设计与实现

摘要： 随着时间的推移，各地对环境条件的要求也在不断提高，对幼苗的需求量也在不断增加，但由于苗源资料的零散，买卖关系的不明确，使得传统的幼苗行业已经无法适应新的形势。本文采用Vscode开发工具结合NodeJS开发了一个基于Vue3的苗木交易平台。系统采用NodeJS 和 Vue3进行开发，应用MVVM的分层思想设计模式以及RESTful API 的Web 服务架构风格。服务端使用Express框架和MongoDB数据库等。平台由客户端和服务端两部分组成，客户端主要包括浏览苗木资源、苗木订购、实时聊天、关注用户、个人中心等功能模块。服务端主要包括统计分析、用户管理、苗木管理、订单管理等功能模块。通过对系统各项功能模块的测试，测试结果良好，满足用户使用需求。

关键词：Vscode；Vue3；NodeJS；Express；MongoDB

Design and Implementation of a Seedling Trading Platform Based on Vue3+NodeJS

Abstract: With the passage of time, the requirements for environmental conditions in various regions are also constantly increasing, and the demand for seedlings is also increasing. However, due to the scattered data of seedling sources and unclear buying and selling relationships, the traditional seedling industry is no longer able to adapt to the new situation. This article uses Vscode development tool combined with NodeJS to develop a seedling trading platform based on Vue3. The system is developed using NodeJS and Vue3, using the hierarchical design pattern of MVVM and the web service architecture style of RESTful API. The server uses the Express framework and MongoDB database. The platform consists of two parts: the client and the server. The client mainly includes functional modules such as browsing seedling resources, ordering seedlings, real-time chat, following users, and personal center. The server mainly includes functional modules such as statistical analysis, user management, seedling management, and order management. Through the testing of various functional modules of the system, the test results are good and meet the user's usage needs.

Keywords: Vscode；Vue3；NodeJS；Express；MongoDB

# 1 绪论

## 1.1 设计背景及意义

近年来通过调查显示，目前传统的大部分苗木产业仍然通过传统的模式运作。传统的苗木交易方式需要大量的人力、物力和时间成本，且存在着信息不对称的问题，使得交易双方难以获得准确的信息。目前我国的线上苗木交易平台有：苗木通、苗木网等，但这些平台基本都是面向大规模苗圃的，对于小规模的个人苗木种植商以及苗木买家并没有提供良好的渠道。因此对于需求量并不大的苗木买家，仍是通过雇佣劳动工人耗时耗力的在散落的村户中，进行逐个的找寻到农户进行交易的。因此基于Vue3 + NodeJS的苗木交易平台的设计和实现具有重要的意义，可以提高苗木交易的效率和质量，推动苗木行业的发展，满足市场需求，具有广阔的应用前景。

## 1.2 设计研究目的

本设计是基于Vue3和 NodeJS的种苗交易平台的设计和实现，苗木交易平台将分散在各个村落中的苗木资源进行聚合，实现苗木资源的信息化，降低需要雇佣劳动工所带来的成本，更快的满足苗木买卖双方的需求。并且通过平台统一的交易，可以提高散户苗木交易的确定性以及安全性，避免了传统运作模式所带来的恶性竞争与交易不确定性。该平台以NodeJS技术为基础进行开发，数据库使用MongoDB，具有更灵活的数据模型以及更高效的读写。满足市场的需求，探索新型苗木交易模型。平台实现用户浏览、收藏、发布苗木资源、关注其他用户，进行苗木订购、在线沟通等功能，并集成地图满足用户大致了解苗木方位的需求，方便用户使用。

## 1.3 本文主要工作和内容

本文主要研究如何实现一个苗木交易平台，本设计主要运用NodeJS语言进行开发，MongoDB作为后台数据库存储数据。首先对系统需求以及可行性进行分析，确定系统的特点以及功能，然后根据功能数据库设计以及进行系统开发。系统实现中需要确定系统开发环境、相关数据连接配置以及核心功能用户注册登陆、关注用户、收藏苗木、在线沟通、苗木交易、苗木资源推荐的实现。展示功能实现的核心代码。然后系统实现后，对其进行系统测试，测试系统运行是否稳定、正常，有无错误。最后进行简单总结，并标注引用到的参考文献。

## 1.4论文的组织方法

本文设计基于Vue3+NodeJS的苗木交易平台，其主要内容安排如下：

第一部分，绪论部分：对苗木交易平台的研究背景和系统的实现意义作简要说明；

第二部分，实现技术介绍：NodeJS语言进行开发、应用MVVM的分层思想设计模式、RESTful API 的Web 服务架构风格、MongoDB数据库；

第三部分，系统分析与设计：对该系统进行详细需求分析和可行性分析，并给出了系统功能模块图和功能用例图；

第四部分，系统实现：给每个模块具体实现进行条理性证明，并粘贴上代码作注释性解释；

第五部分，系统测试：测试该系统的每个功能并总结测试报告；

第六部分，结论：表述完成该毕业设计的收获以及系统还存在的哪些不足。

# 2 相关技术

## 2.1 NodeJS

NodeJS是一个以ChromeV8为基础的 JavaScript运行环境，使 JavaScript不再局限于传统的浏览器，而是直接在服务器上运行，为 JavaScript提供了一种后端开发的功能。它采用事件驱动的非阻塞式 I/O模式，能够对海量的并发请求进行有效的处理，非常适合于构建高性能和可扩展性的网络应用。

## 2.2 Express 服务端框架

Express是一个流行的NodeJS Web应用程序框架，它提供了一组简单易用的API和实现机制，使得开发者可以轻松构建高效、可扩展的Web应用程序。Express具有许多优秀的特性，包括路由、中间件、模板引擎、错误处理等。这些特性可以帮助开发者快速构建Web应用程序，并且具有良好的可扩展性和可维护性。

## 2.3 SocketIO 库

Socket.IO是一个基于NodeJS的实时网络通信库，它允许客户端和服务器之间进行实时双向通信。且他对跨平台支持非常强大，它可以在Web浏览器、移动端、桌面应用程序等多种客户端平台上使用。Socket.IO具有良好的可靠性，支持自动重连和容错机制，可以保证通信的可靠性和稳定性。

2.4 Vue3 框架

Vue3是一个流行的JavaScript框架，用于构建用户界面。它提供了一组简单易用的API和实现机制，使得开发者可以轻松构建高效、可维护的Web应用程序。Vue3具有许多优秀的特性，包括更快的渲染速度、更好的类型支持、更好的开发体验等。Vue3还提供了一些新的特性，例如Composition API、Teleport、Suspense等，使得开发者可以更加灵活地构建应用程序。

2.5 MongoDB 数据库

MongoDB是一种流行的NoSQL数据库，它使用文档数据模型来存储数据，具有高可扩展性、高性能、高灵活性等优点。MongoDB支持复杂的查询和聚合操作，可以处理大量数据，并支持水平扩展，可以轻松地扩展到多个节点。

2.6 vscode 开发环境

Visual Studio Code是一款由微软开发的免费、开源的轻量级文本编辑器和集成开发环境（IDE），它支持多种编程语言，并提供了丰富的扩展和插件来满足不同开发者的需求。具有良好的性能和可扩展性，可以满足从单个文件编辑到大型项目开发的需求。

# 3 需求分析与设计

3.1 业务流程分析

前台功能主要包括浏览苗木资源、查看苗木地址、收藏苗木、关注用户、个人中心、发布苗木资源、实时沟通等功能。本系统前台功能流程图如图3-1所示。

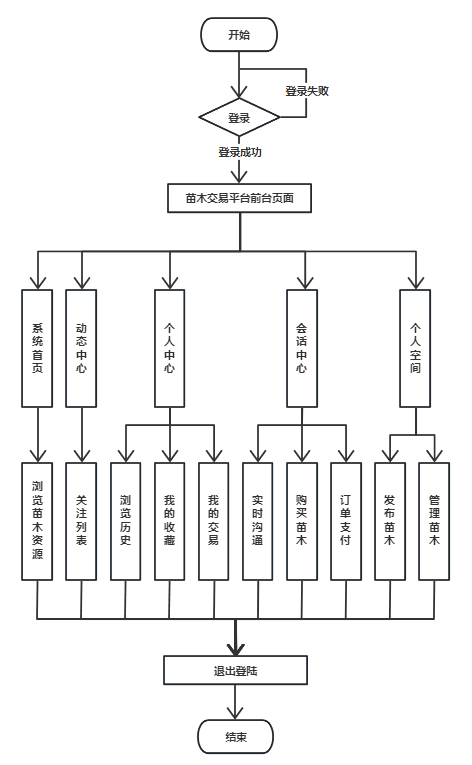


图3-1 系统前台功能流程图

前台部分：

1. 在浏览器中输入网址进入网站首页，进行登录或者注册。
2. 用户可以点击侧边栏导航，选择平台首页、动态中心、个人中心、个人空间、会话中心。

后台功能主要包括统计分析、用户管理、苗木管理、订单管理。本系统后台功能流程图如图3-2所示。

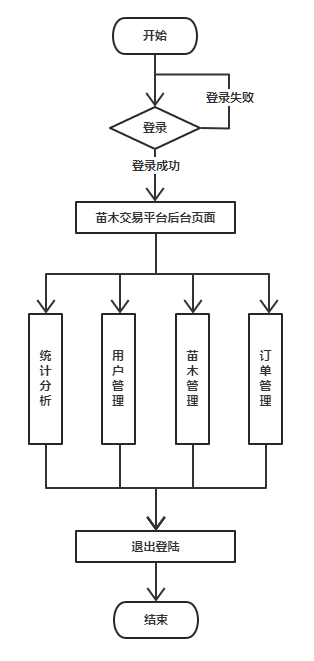


图3-2 系统后台功能流程图

后台部分：

1. 在浏览器中输入网址进入网站首页。
2. 输入账号密码，登录到苗木交易平台的后台，在侧边栏可选择操作对应的功能。
3. 根据相关的功能模块，获取对应的数据进行修改。

## 3.2 系统主要模块与功能的分析概述

《基于Vue3 + NodeJS 的苗木交易平台》是一款真实、安全、透明的平台，以实现苗木资源共享和在线沟通与交易。使用MongoDB作为后台数据库，使用Vue3进行前台页面开发，使用NodeJS作为后台技术，实现页面的展示以及数据交互。通过分析苗木种植商和苗木买家的需求，制定系统的主要功能与模块，最终逐步实现系统。

《基于Vue3 + NodeJS 的苗木交易平台》的主要功能详细描述如下：

1. 前台部分功能模块：
   1. 注册登录模块：游客进入注册页面后，输入必填的信息即可完成注册，用户进入登陆界面，输入账号密码即可完成登录操作。
   2. 首页模块：用户可登陆查看苗木资源，系统根据用户数据可进行推荐苗木资源，也可以选择浏览用户所在地区的苗木资源。
   3. 动态中心模块：用户可登录可查看关注列表以及被关注用户所发布的苗木资源。
   4. 个人中心模块：用户可登录可查看个人浏览历史记录、查看个人收藏的苗木资源、查看个人交易。
   5. 个人空间模块：用户可登录发布苗木资源、管理个人苗木资源、修改个人资料。
   6. 会话中心模块：用户可登录联系用户，进行实时沟通，在线聊天，进行苗木交易。

本系统所有前台功能模块如图3-3所示。

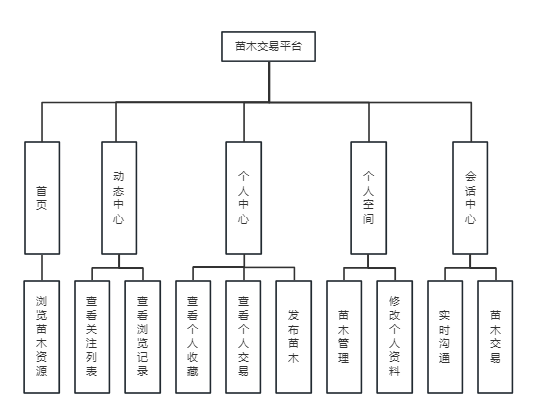


图3-3 前台功能模块图

1. 后台部分功能模块：
   1. 首页模块：管理员可查看统计用户数量、苗木数量、周交易量、以图标的形式展示苗木种类以及每周交易数量。
   2. 用户管理模块：管理员可审核注册用户、修改用户信息。
   3. 苗木管理模块：管理员可发布苗木资源、修改苗木信息、审核用户发布的苗木资源。
   4. 订单管理模块：管理员可查看订单的详细信息，可以查看某个用户的订单信息。

本系统所有后台功能模块如图3-4所示。

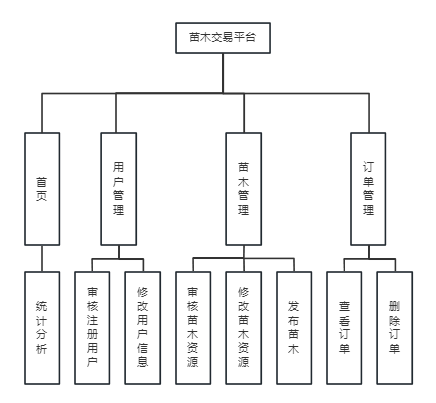


图3-4 后台功能模块图

3.3 系统用例分析

本系统主要包括为普通用户和管理员用户两种用户角色，他们各有不同的用户需求，普通用户需要登录系统、查看苗木资源、对相关苗木进行评论、进行苗木交易、发布苗木资源、关注其他用户；而管理员用户需要能够对用户、苗木信息进行增删改查以及审核操作，查看订单信息以及统计分析的图表。

（1）普通用户需求用例如图3-5所示。

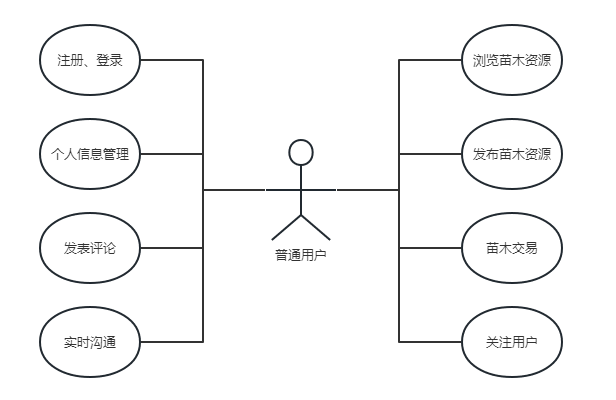


图3-5 普通用户需求用例图

（2）管理员需求用例如图3-6所示。

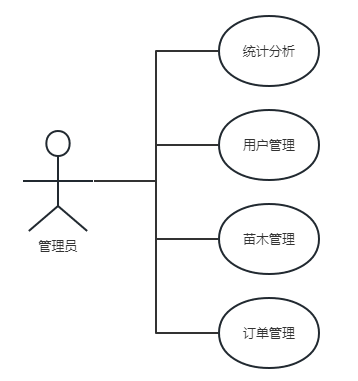


图3-6 管理员需求用例图

## 3.4 系统数据库设计

3.5.1实体关系E-R图

⑴.用户信息实体图如图3-7所示。

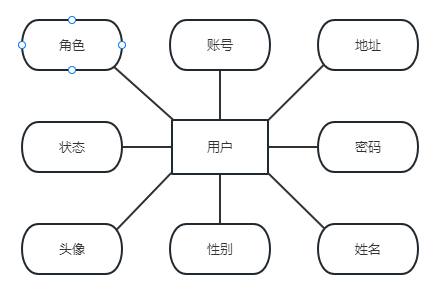


图3-7 用户信息实体图

⑵.苗木信息实体图如图3-5所示。



图3-5 评论信息实体图

⑶.资讯信息实体图如图3-6所示。



图3-6 资讯信息实体图

⑷.课程信息实体图如图3-7所示。



图3-7 课程信息实体图

### 3.5.2 数据表字段设计

本系统采用的是MySQL数据库，是一个关系型数据库管理系统。在本系统中使用到了多张数据表，如资讯表、用户信息表、评论表、预约课程表。具体展示部分如下所示。

表3-4 资讯表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 大小 | 键码 |
| id | int | 11 | 主键 |
| user id | int | 11 | 主键 |
| address | VarChar | 255 | 主键 |
| remark1 | VarChar | 55 | 主键 |
| time | datetime | 60 | 主键 |

表3-5 用户信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 大小 | 键码 |
| id | int | 11 | 主键 |
| username | VarChar | 255 | 主键 |
| pass | VarChar | 255 | 主键 |
| nickname | VarChar | 255 | 主键 |
| sex | VarChar | 255 | 主键 |
| tel | VarChar | 255 | 主键 |
| time | datetime | 60 | 主键 |

表3-6 评论表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 大小 | 键码 |
| id | int | 11 | 主键 |
| pos | VarChar | 255 | 主键 |
| pic | VarChar | 255 | 主键 |
| status | VarChar | 255 | 主键 |

表3-7 预约课程表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 大小 | 键码 |
| id | int | 11 | 主键 |
| user id | VarChar | 55 | 主键 |
| content | text | 35 | 主键 |
| status | VarChar | 255 | 主键 |
| remark1 | VarChar | 255 | 主键 |
| time | datetime | 60 | 主键 |
| targetid | int | 11 | 主键 |

## 3.6本章小结

本章主要通过对本设计的系统需求、系统可行性和系统实现功能进行分析，得出准确的数据以此确定系统具体的功能模块，设计系统数据库。

# 4 系统实现

## 4.1系统开发环境

开发环境需求如下：

（1）硬件要求

CPU：intel I3 2.0GHZ及以上

内存：4GB及以上

硬盘：500G及以上

（2）软件要求

操作系统：win7、win8等操作系统

开发工具：Android Studio、Android SDK

## 4.2 数据库连接配置

在本系统我们是使用Http网络协议技术，它具有GET和POST两种请求方式，对于服务器交互的数据，我们使用JSON字符串，Gson框架对它进行解析和转换，这些方法都分别封装在HttpsUtil.java和JsonUtils.java两个工具类中。具体代码如下。

GET请求方式：

\*http Get请求

\*@param url请求服务器的URL\*@return result返回结果

\*@throws IOException

\*@throws ClientProtocolException\*/

public static String httpDoGet(String url) throws ClientProtocolException,

IOException{

String result = "";

HttpGet request = new HttpGet(url);

request.addHeader("Content-Type", "text/html");request.addHeader("charset", "utf-8");

/获取相应对象

HttpResponse response = new DefaultHIttpClient().execute(request);/当请求成功

if(response.getStatusLine().getStatusCode()==200){  
// 获得响应服务器返回的数据  
result = EntityUtils.toString(response.getEntity());  
}  
return result;  
}

POST请求方式

 \* @param postData 请求的参数  
 \* @param url 请求的url  
 \* @throws UnsupportedEncodingException  
 \* @throws IOException  
 \* @throws KeyManagementException  
 \* @throws NoSuchAlgorithmException  
 \*/  
public static String doPost(String postData, String urlStr)  
throws UnsupportedEncodingException, IOException,  
KeyManagementException, NoSuchAlgorithmException {  
URL url = new URL(urlStr);  
URLConnection cnx = getConnection(url); //连接服务器  
  
OutputStreamWriter wr = new OutputStreamWriter(cnx.getOutputStream());  
wr.write(postData);  
wr.flush();  
wr.close();  
// 取回响应的结果  
return changeInputStream(cnx.getInputStream(), "utf-8");  
}

JSON数据的解析和转换:

 \* @param value  
 \* @return  
 \*/  
public static String createJsonString(Object value) {  
Gson gson = new Gson();  
String string = gson.toJson(value);  
return string;  
}  
  
/\*\*  
 \* JSON解析成对象  
 \* @param jsonString  
 \* @param cls  
 \* @return  
 \*/  
public static <T> T getObject(String jsonString, Class<T> cls) {  
        T t = null;  
        try {  
            Gson gson = new Gson();  
            t = gson.fromJson(jsonString, cls);  
        } catch (Exception e)   
        }  
        return t;  
    }  
/\*\*  
 \* JSON字符串解析成List对象  
 \* @param jsondata  
 \* @param listType  
 \* @return  
 \*/  
public static List<?> StringFromJson(String jsondata, Type listType) {  
Gson gson = new Gson();  
ArrayList<?> list = gson.fromJson(jsondata, listType);  
return list;  
}

## 4.3 系统注册登录

系统注册登录是用户进入该系统并进行体验的第一步，用户首次登入需要进行注册，然后进行登录，其后台是通过视图层（Handler）来接受请求，然后再在该层调用模块层（DAO）查找数据库里的用户信息，来进行注册（写入数据库）或者登陆（查询）的操作。其核心代码如下：  
 login\_login\_text.setOnClickListener {  
 if(TextUtils.isEmpty(login\_username\_edittext.text.toString())){  
 toast("用户名不能为空");  
 return@setOnClickListener;  
 }  
 if(TextUtils.isEmpty(login\_password\_edittext.text.toString())){  
 toast("密码不能为空")  
 return@setOnClickListener;  
 }  
 var map = HashMap<String,String>();  
 map.put("username",login\_username\_edittext.text.toString().trim());  
 map.put("password",login\_password\_edittext.text.toString().trim()); RetrofitTools.post("login",map,User::class.java,object :RetrofitTools.IRetrofitResponse{  
 override fun <T> success(succ: T) {  
 if(TextUtils.isEmpty(succ.toString())){  
 toast("用户名或密码错误");  
 return;  
 }else{  
 var user = succ as User;  
 if(user.userid>0){  
 toast("登录成功")  
 MyApplication.user = user;  
 getToken()  
 }  
 else{  
 toast("用户名或密码错误");  
 }  
 }  
 }  
 override fun failure(msg: String) {  
 toast(msg);  
 }  
 })  
 }  
 }

## 4.4 资讯功能实现

资讯功能模块主要实现的功能是用户可以查看有关摄影师和其摄影作品信息，点击查看详情，进行评论、收藏、预约等操作。其核心代码如下：

TongJi()  
 tongji\_head\_text.setText("资讯详情")  
 var news1=intent.getSerializableExtra("news") as News;  
 news\_detail\_item\_title.setText(news1.title);  
 news\_detail\_item\_contet.setText("\u3000"+news1.content);  
 news\_detail\_item\_contet.setText(Html.fromHtml(news1.content))  
 news\_detail\_item\_time.setText(news1.sendtime);  
 Log.e("TAG",news1.title)  
 }  
}

## 4.5 预约功能实现

预约功能模块主要实现的是用户在找到符合自己要求的摄影师后可以选择进行在线预约，可以选择日期以及拍摄时间，选择完成后，由摄影师根据自己的接单情况和空闲时间确定是否接受该预约，模块实现为YyListActivity，通过setNavBarTitle预约列表，可以显示各种摄影信息以便用户浏览选择。程序核心代码如下所示：

check\_goodsinfo\_cang.setOnClickListener {  
 CircleDialog.Builder(this)  
 .setTitle("提示")  
 .setText("是否确认预约吗")  
 .setPositive("确定", {  
 insertGoods(goods.goodsid.toString())  
 })  
 .setNegative("取消", null)  
 .show()

RetrofitTools.get("insertGoodsOrder",map,String::class.java,object :RetrofitTools.IRetrofitResponse{  
 override fun <T> success(succ: T) {  
 if("true".equals(succ)){  
 toast("预约成功")  
 }else{  
 toast("你已经预约过了")  
 }  
 }  
 override fun failure(msg: String) {  
 toast(msg)  
 }  
4.6 评论功能实现

class GoodsInfoActivity : BaseActivity() {  
 lateinit var goods:Goods;  
 var date = ""  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_goods\_info)  
 TongJi()  
 tongji\_head\_text.setText("详细")  
 goods = intent.getSerializableExtra("goods") as Goods  
 Glide.with(this).load(MyApplication.hostFile+goods.pic).error(R.mipmap.ic\_launcher).into(goodsinfo\_detail\_iv)  
 goodsinfo\_detail\_stock.setText("${goods.title}")  
 goodsinfo\_detail\_content.setText(goods.content)  
 goodsinfo\_detail\_price.setText("${goods.price}")  
 //我要评论  
 goodsinfo\_discuss.setOnClickListener {  
 CircleDialog.Builder(this@GoodsInfoActivity)  
 .setTitle("评论")  
 .setInputHint("请输入评论内容")  
 .setPositiveInput("确定",object: OnInputClickListener {  
 override fun onClick(text: String?, v: View?) {  
 if (TextUtils.isEmpty(text)){  
 toast("内容不能为空")  
 return;  
 }  
 insertData(goods.goodsid.toString(),text!!)  
 }  
 })  
 .setNegative("取消", null)  
 .show()  
 }

## 4.6用户管理

登录用户管理模块后，用户可以在该处查看个人信息，当用户有需要时可以对用户个人信息增删改查，如果是管理员登录用户管理模块，除了可以查看用户信息，还可以查看系统预约情况，评论情况，摄影信息发布情况等。核心代码如下。

TongJi()  
 tongji\_head\_text.setText("个人信息")  
 setContentView(R.layout.activity\_show\_person)  
 show\_person\_username\_tv.setText(MyApplication.user.username)  
 show\_person\_nickname\_tv.setText(MyApplication.user.nickname)  
 show\_person\_age\_tv.setText(MyApplication.user.age.toString())  
 show\_person\_tel\_tv.setText(MyApplication.user.tel)  
 show\_person\_address\_tv.setText(MyApplication.user.address)  
 show\_person\_nickname\_tv.setOnClickListener **{** CircleDialog.Builder(this@ShowPersonActivity)  
 .setTitle("昵称")  
 .setInputHint("请输入昵称")  
 .setPositiveInput("确定",object: OnInputClickListener {  
 override funonClick(text: String?, v: View?) {  
 if (TextUtils.isEmpty(text)){  
 toast("内容不能为空")  
 return;  
 }  
 getdata("nickname",text!!)  
 MyApplication.user.nickname= text;  
 show\_person\_nickname\_tv.setText(text!!)  
 }  
 })  
 .setNegative("取消", null)  
 .show()  
 **}** show\_person\_address\_tv.setOnClickListener **{** CircleDialog.Builder(this@ShowPersonActivity)  
 .setTitle("地址")  
 .setInputHint("请输入地址")  
 .setPositiveInput("确定",object: OnInputClickListener {  
 override funonClick(text: String?, v: View?) {  
 if (TextUtils.isEmpty(text)){  
 toast("内容不能为空")  
 return;  
 }  
 getdata("address",text!!)  
 MyApplication.user.address = text;  
 show\_person\_address\_tv.setText(text!!)  
 }  
 })  
 .setNegative("取消", null)  
 .show()  
 **}** show\_person\_tel\_tv.setOnClickListener **{** CircleDialog.Builder(this@ShowPersonActivity)  
 .setTitle("电话")  
 .setInputHint("请输入电话")  
 .setPositiveInput("确定",object: OnInputClickListener {  
 override funonClick(text: String?, v: View?) {  
 if (TextUtils.isEmpty(text)){  
 toast("内容不能为空")  
 return;  
 }  
 getdata("address",text!!)  
 MyApplication.user.tel= text;  
 show\_person\_tel\_tv.setText(text!!)  
 }  
 })  
 .setNegative("取消", null)  
 .show()  
 **}** show\_person\_age\_tv.setOnClickListener **{** CircleDialog.Builder(this@ShowPersonActivity)  
 .setTitle("年龄")  
 .setInputHint("请输入年龄")  
 .setPositiveInput("确定",object: OnInputClickListener {  
 override funonClick(text: String?, v: View?) {  
 if(TextUtils.isEmpty(text)){  
 toast("内容不能为空")  
 return;  
 }  
 getdata("age",text!!)  
 MyApplication.user.*age* =text .*toInt*();  
 show\_person\_age\_tv.setText(text!!)  
 }  
 })  
 .setNegative("取消", null)  
 .show()  
 **}** }

## 4.7评论功能实现

评论功能模块实现的主要功能是用户可以在浏览摄影作品后在留言区留下对该作品的看法，表达自己的想法和摄影师进行交流，或者是在客户在结束拍摄后以自己的拍摄体验为基础给摄影师评价留言；核心代码如下：

goodsinfo\_discuss.setOnClickListener {  
 CircleDialog.Builder(this@GoodsInfoActivity)  
 .setTitle("评论")  
 .setInputHint("请输入评论内容")  
 .setPositiveInput("确定",object: OnInputClickListener {  
 override fun onClick(text: String?, v: View?) {  
 if (TextUtils.isEmpty(text)){  
 toast("内容不能为空")  
 return;  
 }  
 insertData(goods.goodsid.toString(),text!!)  
 }  
 })  
 .setNegative("取消", null)  
 .show()  
 }

## 4.8本章小结

本章主要叙述网上摄影预约管理系统实现所需要的开发环境，展示数据库连接配置数据以及系统主要功能模块的具体实现核心代码。

# 5 系统测试

5.1测试概述

系统测试是一种非常常见的查错方法，通过系统测试能够检测系统的问题降低系统错误率。系统开发的整个工程巨大，在开发过程中可能会存在很多我们没注意到的错误，通过系统测试完整检测系统的工作状态和程序代码运行情况。确保系统功能完整实现。通常系统测试包括:安全性测试、性能测试、可用性测试、功能测试、兼容性测试等。

## 5.2测试方案

本系统采用的测试方案是黑盒测试法，就是直接测试该系统的各个功能，检测是否有漏洞影响用户使用。顾名思义，这种测试方法把整个网上摄影预约管理系统当做一个被封装的盒子，而不去深究系统内部的逻辑和业务，但其和集成测试也有所区别，只对程序封装接口进行测试，测试是否能够按照需求说明书进行使用。

## 5.3 测试用例

本系统每个功能模块都能正常运行切具有一定的容错性，在这里选取以下表功能模块进行测试用例说明：

用户登录：测试系统登录功能是否正常、稳定。如表5-1所示；

用户注册：测试系统用户注册功能是否正常、稳定。如表5-2所示；

评论模块：测试评论模块功能是否正常、稳定如表5-3所示；

表5-1 用户登录测试表

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 用户登录 | 测试序号 | 01 |
| 测试时间 | 2022年04月15日 | 测试人员 | 开发者 |
| 测试目的 | 测试系统登录功能是否正常、稳定 | | |
| 测试步骤 | （1）用户要登录本系统并在登录界面输入用户名“1234”，密码“1234”点击登录按钮  （2）用户要登录本系统并在登录界面输入用户名“1234”，密码“123”点击登录按钮 | | |
| 预测结果 | （1）通过后台服务器与数据库的信息匹配正确后，提示“登录成功”，并进入主界面，用户1234可以正常登录系统。  （2）经过客户端代码对输入的数据进行格式规范校验后，提示“用户名或密码不对，请重新输入”。 | | |
| 实际结果 | （1）通过后台服务器与数据库信息匹配正确后，提示“登录成功”，并进入主界面，用户1234可以正常登录系统。  （2）经过客户端代码对输入的数据进行格式规范校验后，提示“用户名或密码不对，请重新输入” | | |

表5-2 用户注册测试表

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 用户注册 | 测试序号 | 02 |
| 测试时间 | 2022年04月15日 | 测试人员 | 开发者 |
| 测试目的 | 测试系统用户注册功能是否正常、稳定 | | |
| 测试步骤 | （1）输入“用户名”、“密码”，“手机号”、“邮箱”等信息，并点击注册按钮。  （2）直接点击注册按钮。 | | |
| 预测结果 | （1）系统将注册信息验证后录入数据库，并提示“注册成功”，跳转到登录界面。  （2）注册失败，提示“请输入用户名” | | |
| 实际结果 | （1）系统将注册信息验证后录入数据库，并提示“注册成功”，跳转到登录界面。  （2）注册失败，提示“请输入用户名”。 | | |

表5-3评论模块测试表

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 评论模块 | 测试序号 | 03 |
| 测试时间 | 2022年04月15日 | 测试人员 | 开发者 |
| 测试目的 | 测试评论模块功能是否正常、稳定 | | |
| 测试步骤 | 在评论模块界面，点击发布帖子，进行评论、点赞等操作。 | | |
| 预测结果 | 操作成功。 | | |
| 实际结果 | 操作成功。 | | |

## 5.4 测试结果分析

### 5.4.1系统注册登录测试结果

用户通过登录进入可视化系统，登录时会校验用户名密码，如果密码或者验证码错误系统给出提示，如果正确则成功的进入系统，其登录成功跳转到首页。

（1）用户登录成功的测试结果如图5-1所示：

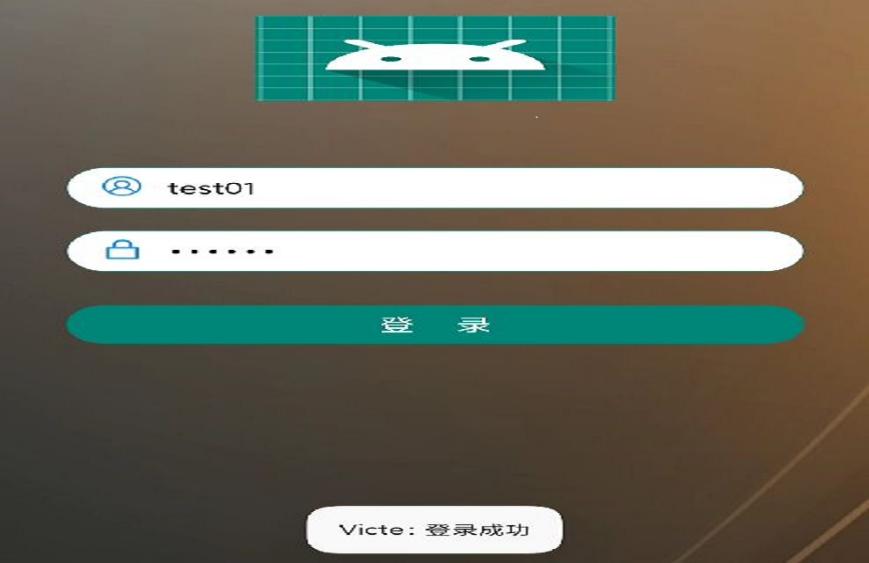


图5-1 系统注册登录测试图

（2）用户登录成功跳转到首页的测试结果如图5-2所示：



图5-2 用户登录成功跳转到首页测试图

5.4.2资讯管理测试图

用户在登入系统后可以进行搜索自己喜欢的摄影风格，浏览不同的摄影作品，并在作品下方的留言区进行评论，和摄影师交流摄影作品。咨询管理预约管理测试结果如图5-3所示：

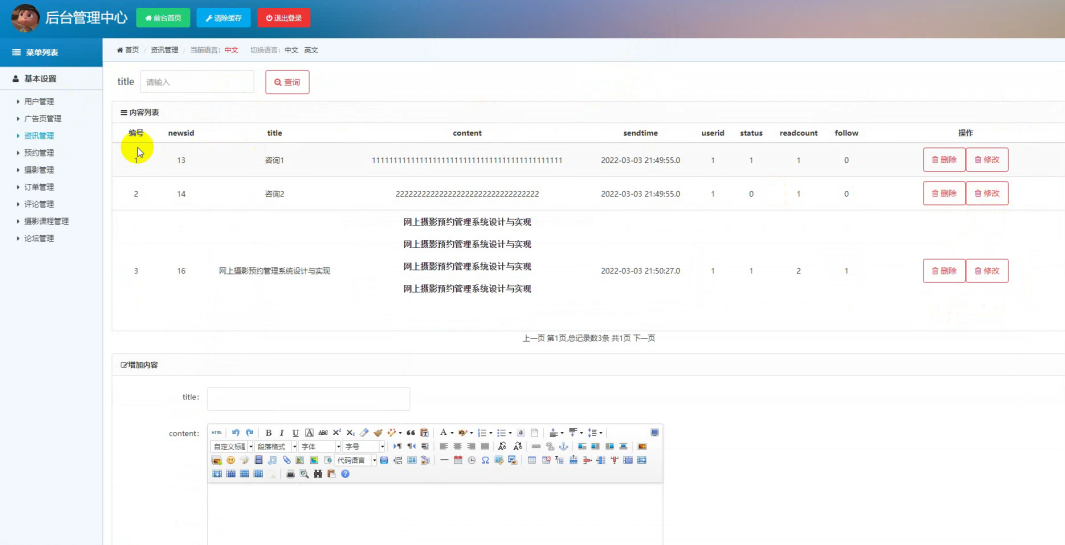


图5-3 资讯管理测试图

### 5.4.3预约管理模块测试图

预约管理模块是本系统非常重要的模块，在用户选择了摄影师后进行在线预约，可以选择日期以及拍摄时间，选择完成后，在线提交预约请求。

1. 预约成功测试结果如图5-4所示：



图5-4 预约成功测试图

（2）预约管理模块测试结果如图5-5所示

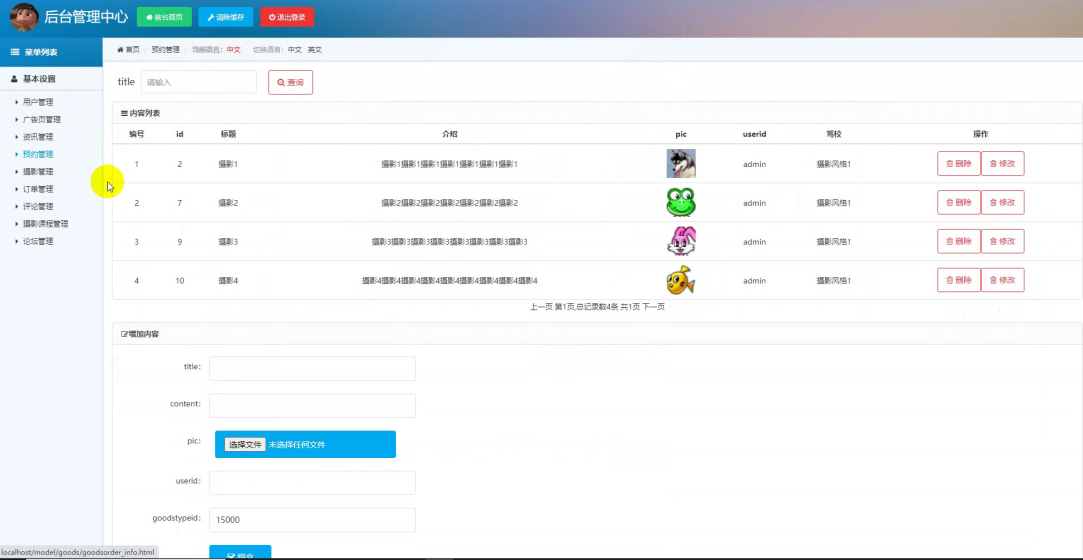


图5-5 预约管理模块测试图

### 5.4.4用户管理模块测试结果

用户管理测试是用户可以通过后台管理中心查看用户个人信息，对用户信息进行修改，如用户id、昵称、头像和密码等。测试结果如图5-6所示：

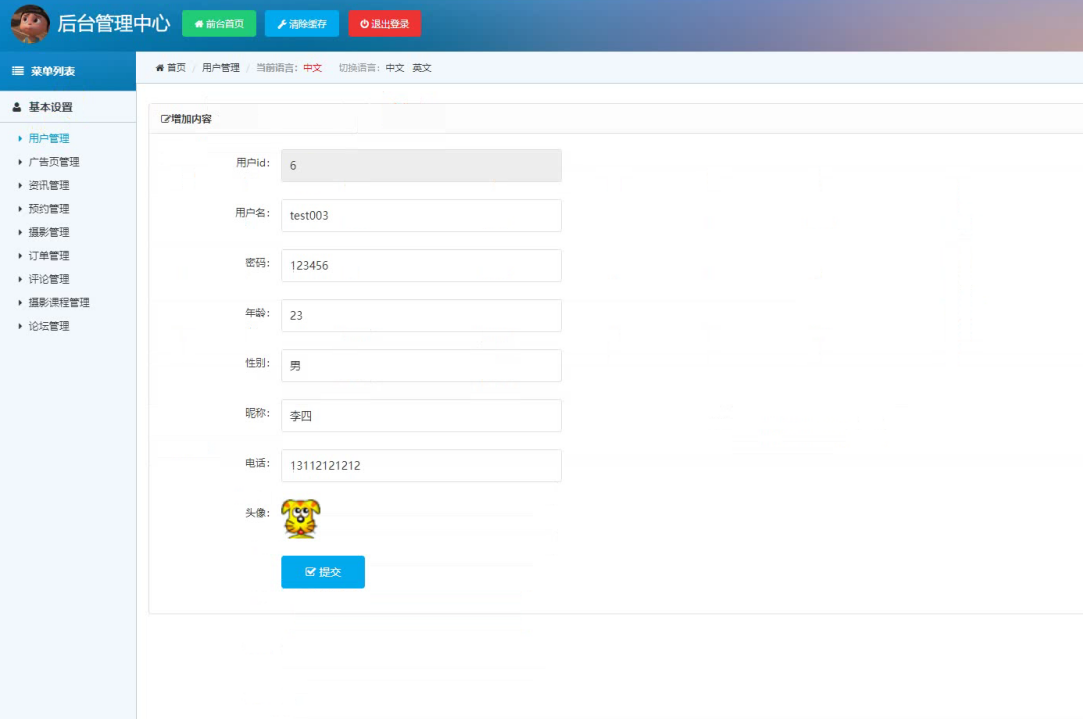


图5-6 用户管理模块测试图

5.4.5评论管理模块测试结果

评论管理模块测试是用户在该部分对摄影作品留言评论并查看自己的评论记录，也可进行删除修改。评管理模块测试结果如图5-7所示：

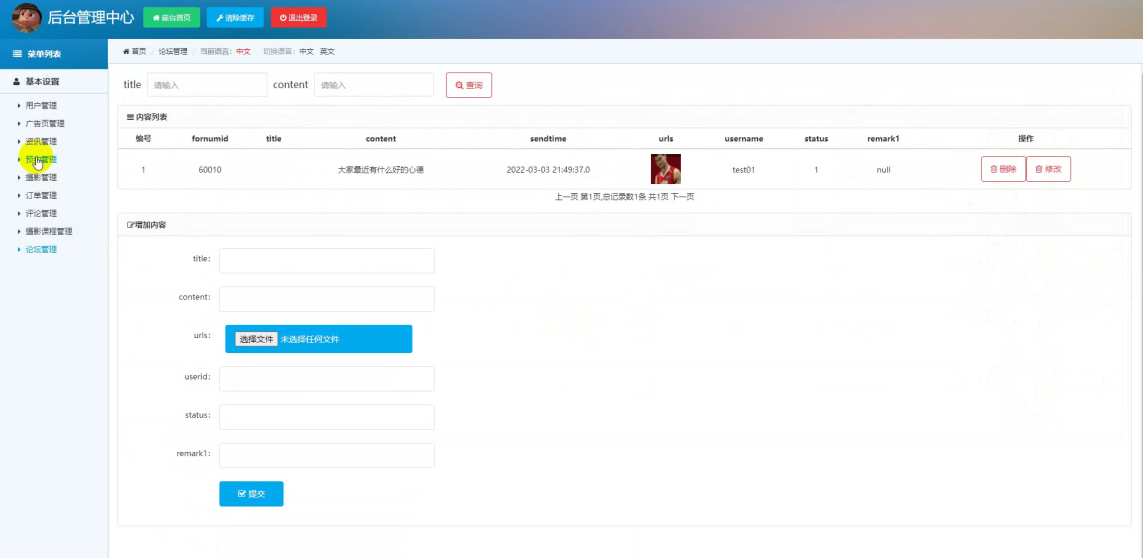


图5-7 课程管理模块测试图

5.5本章小结

本章主要叙述系统测试的相关内容，系统测试选择黑盒测试法，通过选取用户注册登录和用户评论部分展示系统测试流程和内容。接下来展示系统主要功能模块如注册登录模块、资讯管理模块、预约管理模块、用户管理模块、评论管理模块的测试结果图，证明该系统的实现。

# 结束语

本文主要阐述了基于Android操作系统结合Android Studio开发工具开发的基于的网上摄影预约管理系统的设计与实现。系统运用Java语言进行开发，使用IDEA作为Java编程语言开发的集成环境，用户界面基于Android的B/S结构，应用框架设计模式采用MVC的分层思想，后台设计使用tomcat服务器和MySQL数据库。系统主要实现用户管理、课程管理、资讯管理、预约管理、评论管理等功能。虽然本设计的基本功能已经实现但由于自己的本身能力不足，系统还是有很多地方需要加以完善，比如系统的页面制作粗糙，美观不够，而且很功能也比较简单，希望在以后的时间里可以再继续加以改进。

# 参考文献

[1] 郭宏志. Android应用开发详解［M］.北京: 电子工业出版社 2016.6 417-420.

[2] 马颜军. 基于Android系统的APP开发[J]. 科技与企业, 2015(22):87-88.

[3] 叶宾. Android系统的手机软件开发研究[J]. 中国新技术新产品, 2015(10):12-13.

[4] 王子榕. 浅谈基于安卓操作系统的应用软件开发[J]. 通讯世界, 2015(14):220-220.

[5] 詹凯. 基于安卓操作系统的应用软件开发分析[J]. 通讯世界, 2016(7):262-263.

[6] 刘鹏. 基于Android系统的手机软件开发研究[J]. 信息与电脑:理论版, 2016(9):95-96.

[7] 周宇, 尹生强, 王冬青,等. 基于Android Studio和Android系统的App开发平台搭建研究[J]. 青岛大学学报(工程技术版), 2016, 31(3):49-53.

[8] 肇宇飞, 娄宝文, 王翔. 基于Android开发的车联网APP设计[J]. 山东工业技术, 2016(18):134-134.

[9] 陈婷. 基于Android的移动学习系统研发[D]. 华南理工大学, 2015.

[10] 田兴强. Android的若干关键技术的研究及其应用系统的开发分析[J]. 电脑知识与技术:学术交流, 2015, 11(11):77-78.

[11] 宋春晖, 廉东升. 安卓系统应用与开发研究[J]. 电子测试, 2016(15):82-82.

[12] 王勇, 赵丽红, 杨四化,等. 基于Android系统开发的触摸屏用户交互的场馆计分系统[J]. 电子世界, 2017(14):133-133.

[13] 潘港超, 焦佳彭, 耶萍. 基于Android系统的校园电子商务平台开发[J]. 电脑编程技巧与维护, 2015(3):38-39.

[14] 赵静. Android系统架构及应用程序开发研究[J]. 自动化与仪器仪表, 2017(1):86-87.

[15] 赵晓伟. Android 开发环境在 Linux 平台上的搭建[J]. 计算机与数字工程, 2016, 44(8):1615-1618.

[16]周鹏. 基于Android平台和OBD-Ⅱ的车联网应用系统设计与开发[D]. 重庆大学, 2016.

[17] Erika, Porter A, Greenwood, et al. Analyzing inter-application communication in Android[J]. 2015.

[18] Faruki P, Bharmal A, Laxmi V, et al. Android Security: A Survey of Issues, Malware Penetration, and Defenses[J]. IEEE Communications Surveys & Tutorials, 2015, 17(2):998-1022.

[19] Zhauniarovich Y, Ahmad M, Gadyatskaya O, et al. StaDynA: Addressing the Problem of Dynamic Code Updates in the Security Analysis of Android Applications[C]// ACM Conference on Data and Application Security and Privacy. ACM, 2015:37-48.

# 致谢

本次毕业设计对我来说就是大学四年学习的凝练，在这次毕业设计过程中我遇到了各种各样的阻碍和困难，但值得庆幸的是，在我的指导老师—赵瑞斌和张燕玲老师以及周围同学的帮助下我终于完成了此次毕业设计，给自己的大学生活画上一个句号。

本次毕业设计主要使用的Android操作系统，还有IDEA开发工具以及MySQL数据库虽然在大学四年中都是课上学习过的知识，但是在真正使用它们做毕业设计时还是感觉理论和实际操作有很大的差距，虽然理解理论如何应用，但是在动手操作时有很大障碍。因此，在做毕业设计时，我是一边进行系统开发，一边也在学习相关的知识。都说前人栽树，后人乘凉，非常感谢前辈们给我们提供了很多便利，让我们在这条道路上走的轻松很多。在遇到自己无法解决的问题时我就会在各种网站上寻找解决办法，其中CSDN网站就非常的适合我们，CSDN上面可以找到很多我们需要的东西同时有很多行业前辈，我们可以向他们寻求帮助和他们交流请教问题解决办法。但由于自己的本身能力不足，系统还是有很多地方需要加以完善，比如系统的页面制作粗糙，美观不够，而且很功能也比较简单，如果以后有机会，希望可以再继续加以改进。

在此我再次感谢我的毕业设计指导老师赵卫斌和张艳玲老师，如果没有他们的倾心帮助，我的毕业设计肯定会遇到更多的困难。