

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 系统开发环境介绍

## 2.1 Android平台基础知识

Android是一个应用非常广泛的智能手机操作系统和软件平台，比较多的是使用在便捷式移动设备上。Android系统的结构框架采用了从高层到底层共四层的分层框架，分别是应用程序层（Applications）、应用程序框架层（Applications Framework）、系统运行库层（Libraries和Android Runtime）和Linux内核层（Linux Kernel）。其中，第一层的应用程序层中所有的应用程序都是用Java语言编写的，各种与用户直接交互的应用程序，如SMS短信客户端、图库和Web浏览器等。应用程序框架层为开发者在应用程序层的开发提供了API，使开发人员能够进行快速有效的应用程序开发。系统运行库是Android运行构成的，专门为Android系统中各个组件提供服务的。Android以Linux操作系统内核为基础，在Linux内核层实现硬件设备驱动、进程和内存管理、网络协议栈、电源管理和无线通信等功能。本系统是在Android的应用程序层上开发的，应用涉及到了应用程序框架层的四大组件、Intent与Intent Filter等相关技术，还有使用Android的五大布局对用户界面设计。

## 2.2 Android用户界面简介

Android是一个应用非常广泛的智能手机操作系统和软件平台，比较多的是使用在便捷式移动设备上。Android系统从结构框架上看，它是采用了分层的框架，从高层到底层分为四层，分别是应用程序层（Applications）、应用程序框架层（Applications Framework）、系统运行库层（Libraries和Android Runtime）和Linux内核层（Linux Kernel）。其中，应用程序层包含了用Java语言编写的，各种与用户直接交互的应用程序，如SMS短信客户端、图库和Web浏览器等。应用程序框架层为开发者在应用程序层的开发提供了API，使开发人员能够进行快速有效的应用程序开发。系统运行库是Android运行构成的，专门为Android系统中各个组件提供服务的。Android以Linux操作系统内核为基础，在Linux内核层实现硬件设备驱动、进程和内存管理、网络协议栈、电源管理和无线通信等功能。本系统是在Android的应用程序层上开发的，应用涉及到了应用程序框架层的四大组件、Intent与Intent Filter等相关技术，还有使用Android的五大布局对用户界面进行设计。上述设计的技术具体介绍如下。

（1）Android四大组件：Activity（活动）、Service（服务）、Broadcast Receiver（广播接收者）、Content Provider（内容提供者）；

Activity：负责应用程序界面显示最基本的Android组件，人机交互的基本单元。具有运行时的上下文和独特的生命周期，用户看到的每个界面就代表一个Activity，每个Activity都是android app Activity 的子类，在Activity的对应的生命周期方法中做相应的逻辑处理。在Android中它有4种启动模式，分别为standard（默认）、singleTop、singleTask、singleInstance。

Service：是在后台执行某一个长期任务的组件，它不包含人机交互的界面，可以支持如网络事务、音乐播放、后台文件下载等任意类型的后台任务。总的来说Service组件有两种使用方法：第一种用法是实现绑定IPC（进程间通信）的接口，从而实现支持远程调用。另一种用法则是通过应用程序的内部组件触发的，启动的方式有两种分别是startService()与bindService()。

Broadcast Receiver：是用于接收系统内部范围内的广播产品消息，并做出对应处理的组件，在我看来它还可以用来作为Android组件之间、应用程序之间的数据传递。它本身也没有与用户交互的界面，所以一般来说可以启动一个Activity来响应接收到的消息，或者用NotificationManager来产品用户。

Content Provider：是Android系统数据共享机制的组件，主要用来实现应用之间的数据共享。系统本身也自带该共享机制，如共享本地的音乐、音频，图片等。

（2）Intent 和 Intent Filter：

Intent：顾名思义就是是“意图”的意思，Intent机制是Android中用来协助应用间以及组件之间的交互通讯，通过Intent来执行一次操作的动作和组件之间的数据传递。

Intent Filter：就是 Intent的过滤器，一般只应用在隐式Intent中，如注册Broadcast Receiver和Service的时候就使用到。而它应用在显式Intent中是不起作用的

（3）Android的五大布局，分别是FrameLayout（帧布局）、LinearLayout（线性布局）、AbsoluteLayout（绝对布局）、RelativeLayout（相对布局）。

## 2.3 开发工具介绍

### 2.3.1 Android Studio介绍

Android Studio是一个Android集成开发工具，基于IntelliJIDEA，Android Studio提供了集成的Android开发工具用于开发和调试。开发者可以在编写程序的同时看到自己的应用在不同尺寸屏幕中的样子。

### 2.3.2 关键技术介绍

Android平台开发方面包含基本的应用功能开发、数据存储、网络访问这三大块。



① 应用功能开发

标准的[Android](http://dev.10086.cn/cmdn/wiki/index.php?doc-view-1789)程序由如下4部分组成即Activity、Broadcast Intent Receiver、Service、Content Provider来实现。

② 数据存储

在Android，可供选择的存储方式包括了SharedPreferences、文件存储、SQLite数据库存储方式、内容提供器方式(Content Provider)以及网络方式 5种，本系统网络方式主要是通过网络访问该网络提供的网络服务接口实现数据的读写服务。

③ 网络访问方面

主要是Http访问技术的封装，通过java[.NET](http://dev.10086.cn/cmdn/wiki/index.php?doc-view-1452).\*；以及Android.net.\*；下面提供的HttpPost、DefaultHttpClient、HttpResponse等类提供的访问接口来实现具体的[web](http://dev.10086.cn/cmdn/wiki/index.php?doc-view-1500)服务访问。

本软件采用Java作为开发语言，布局和数据存储是开发本软件的关键技术，布局用来设计用户交互界面，使用xml技术设计，而xml中控件的位置和状态则离不开布局的使用，用户界面的设计是否美观和合理直接影响用户体验，因此运用好布局技术是直接和用户相关的，而数据存储技术则记录了用户所有的操作数据和用户所需要提取的数据。

2.4本章小结

本章主要介绍为实现网上摄影预约管理系统所用到的开发环境及开发工具的相关知识。系统基于Android操作系统实现，开发工具选择Android Studio，基于IDEA，为实现系统功能提供理论工具保障。

# 3 系统分析与设计

3.1需求分析

网上摄影预约系统主要目的是汇集大量摄影师的摄影信息存放在数据库，并开发评论、预约功能，用户在登录平台后可以直接搜索选择符合自己喜好的摄影师进行预约操作。并且在结束拍摄后可以在平台的论坛里进行讨论交流，给摄影师进行评价。

## 3.2 可行性分析

### 3.2.1 技术可行性

如今的Android智能手机性价比高、价格优惠、系统升级快加上无处不在的移动网络，性能和硬件方面都是可以满足软件运行的需求的。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数据库等，这些开发工具和技术都是现在正在被广泛应用的。所以，开发本系统在技术上是可行的。

### 3.2.2 操作可行性

本系统采用的Android操作系统和Android Studio开发工具都是目前很常见的开发系统所能使用的，应用简单很容易熟练掌握。且本系统的界面设计简单易操作，用户通过智能手机和电脑端都可以操作访问。

### 3.2.3 经济可行性

在本系统开发所使用的开发工具android是开源性操作系统，开发环境的搭建都是开源免费的都不需要用户花钱购买，所以也不存在经济问题，

3.3 系统用例分析

本系统主要包括为普通系统登陆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，普通系统登陆用户需要登录系统、查找咨询摄影信息、能够在评论区进行留言评论、选择摄影师预约；而管理员用户需要能够对用户个人信息进行增删改查操作，查看资讯、预约、评论内容。

（1）用户需求用例如图3-1所示。



图3-1 用户需求用例图

（2）管理员需求用例如图3-2所示。



图3-2 管理员需求用例图

（3）课程管理用例分析表

表3-1 课程管理用例分析

|  |  |
| --- | --- |
| 用例名称  用例简介  前置条件 | 课程管理  对课程信息进行添加、删除等管理。  系统登录正常，网络连接正常。 |
| 主要操作 | 添加课程明细  （1）管理员打开系统。  （2）进入系统登录界面，输入设定好的登录账号及密码。  （3）登录系统，进入系统主界面。  （4）点击课程管理功能，进入课程管理界面。  （5）点击课程管理界面的添加课程信息按钮，进入添加课程信息操作界面。  （6）按提示，选择或输入名称、课时、发布时间等信息。  （7）确认信息正确后，点击确定按钮，确认添加，课程信息添加入数据库中。  删除课程信息明细  （1）处于课程管理界面。  （2）点击想要删除的课程信息条目后的删除按钮，弹出课程信息删除确认界面。  （3）点击确认按钮，实现课程信息的删除，相应课程信息从数据库中删除。 |

（4）资讯管理用例分析表

表3-2 资讯管理用例分析

|  |  |
| --- | --- |
| 用例名称  用例简介  前置条件 | 资讯管理  对资讯信息进行添加、删除等管理。  系统登录正常，网络连接正常。 |
| 主要操作 | 添加资讯明细  （1）管理员打开系统。  （2）进入系统登录界面，输入设定好的登录账号及密码。  （3）登录系统，进入系统主界面。  （4）点击资讯管理功能，进入资讯管理界面。  （5）点击资讯管理界面的添加资讯信息按钮，进入添加资讯信息操作界面。  （6）按提示，选择或输入标题、内容、发布时间等信息。  （7）确认信息正确后，点击确定按钮，确认添加，资讯信息添加入数据库中。  删除资讯信息明细  （1）处于资讯管理界面。  （2）点击想要删除的资讯信息条目后的删除按钮，弹出资讯信息删除确认界面。  （3）点击确认按钮，实现资讯信息的删除，相应资讯信息从数据库中删除。 |

（5）用户管理用例分析表

表3-3 用户管理用例分析

|  |  |
| --- | --- |
| 用例名称  用例简介  前置条件 | 用户管理  对用户信息进行添加、删除等管理。  系统登录正常，网络连接正常。 |
| 主要操作 | 添加用户明细  （1）管理员打开系统。  （2）进入系统登录界面，输入设定好的登录账号及密码。  （3）登录系统，进入系统主界面。  （4）点击用户管理功能，进入用户管理界面。  （5）点击用户管理界面的添加用户信息按钮，进入添加用户信息操作界面。  （6）按提示，选择或输入姓名、头像、电话等信息。  （7）确认信息正确后，点击确定按钮，确认添加，用户信息添加入数据库中。  删除用户信息明细  （1）处于用户管理界面。  （2）点击想要删除的用户信息条目后的删除按钮，弹出用户信息删除确认界面。  （3）点击确认按钮，实现用户信息的删除，相应用户信息从数据库中删除。 |

## [3.4 系统功能分析](#_Toc137272364)

本系统是给想要寻找摄影师为自己拍摄照片的用户提供摄影师相关信息的Web应用，具有信息收集、信息分析和信息展示等功能，最大程度满足用户对摄影师信息便捷获取的需求。具体的系统功能设计模块图如图3-1所示：

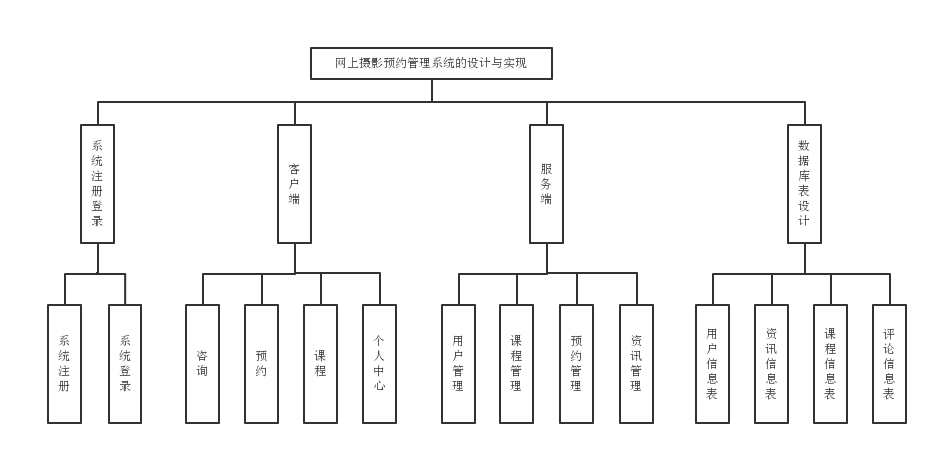


图3-3 系统功能模块图

需求分析就是分析用户的需要与要求，在功能和性能上有描述。它是软件开发过程中的第一阶段的一部分，主要任务是要了解你需要什么，什么应做的工作，以确定系统的，因此在目标系统提出完整，准确，清晰，具体的要求，并表示以书面形式唤醒。

系统由客户端和服务端两部分组成，客户端主要包括资讯、预约、课程、论坛、个人中心等功能模块。服务端主要包括用户管理、课程管理、资讯管理、预约管理、评论管理、论坛管理等功能模块。

## 3.5 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3-4所示。



图3-4 用户信息实体图

⑵.评论信息实体图如图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：

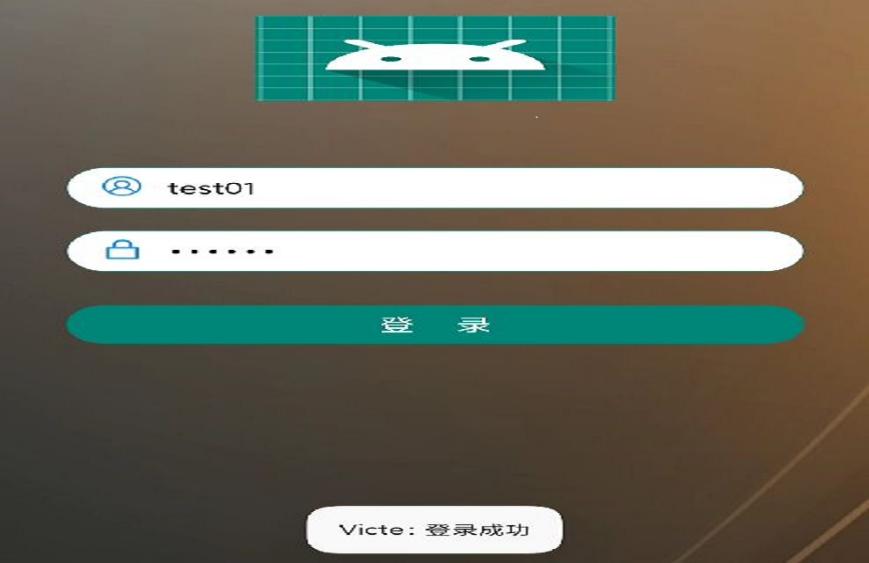


图5-1 系统注册登录测试图

（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：

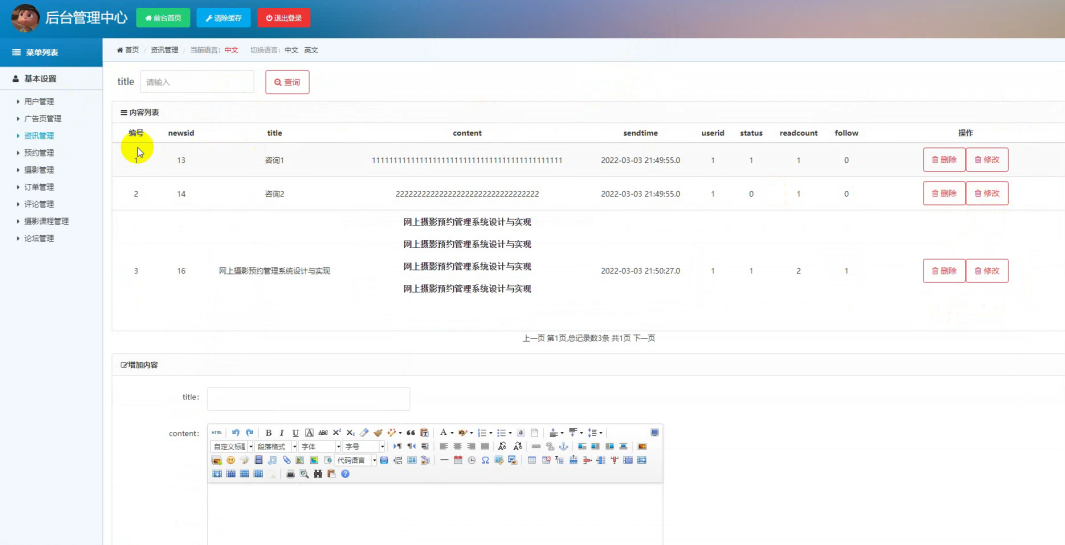


图5-3 资讯管理测试图

### 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

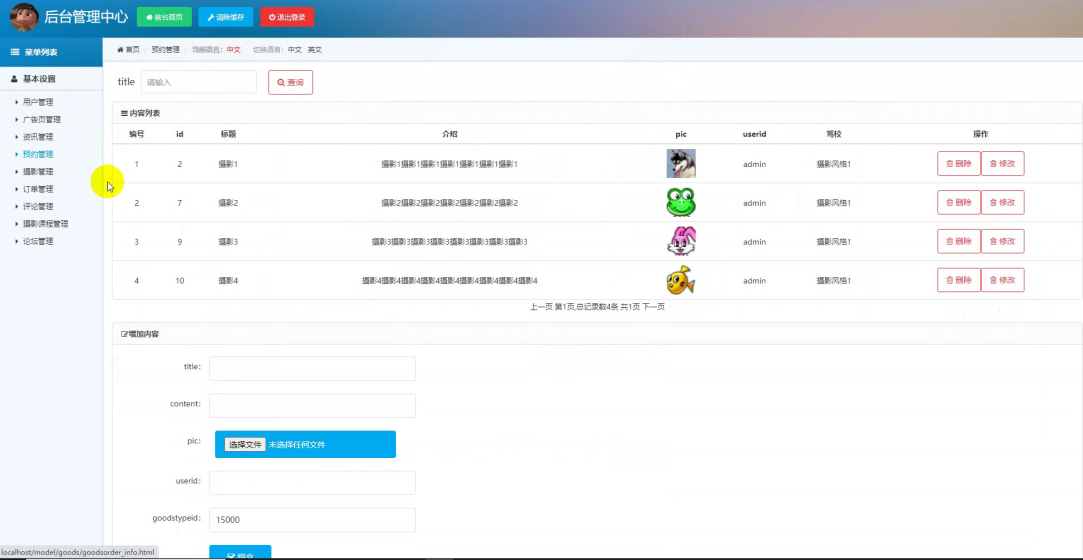


图5-5 预约管理模块测试图

### 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：

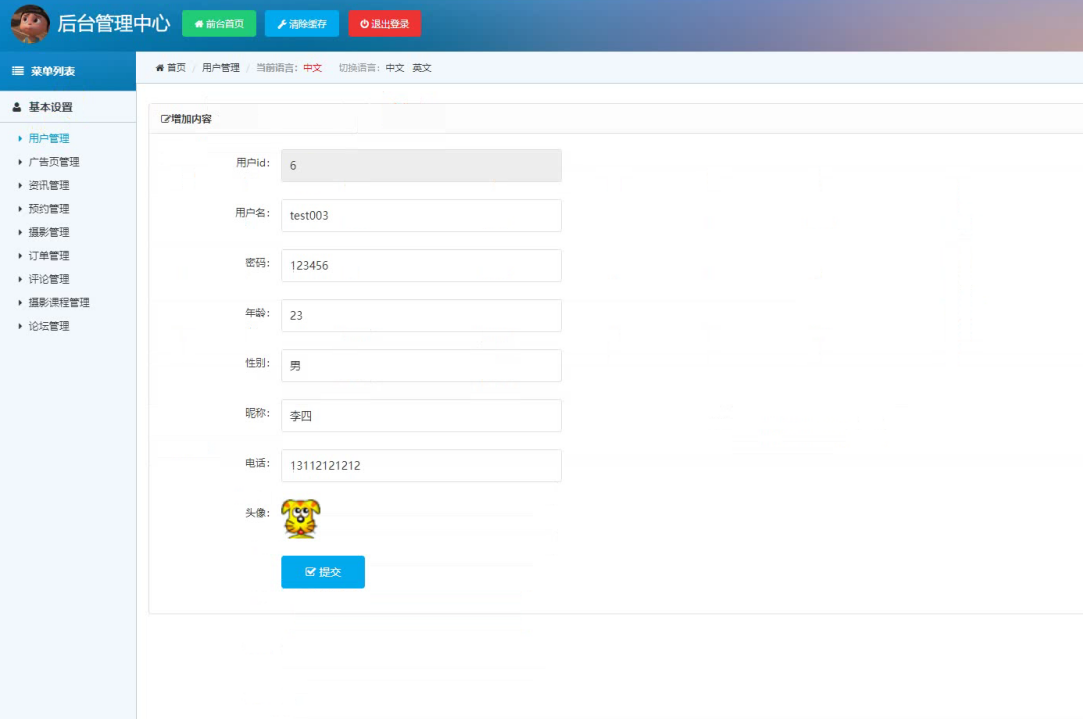


图5-6 用户管理模块测试图

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：

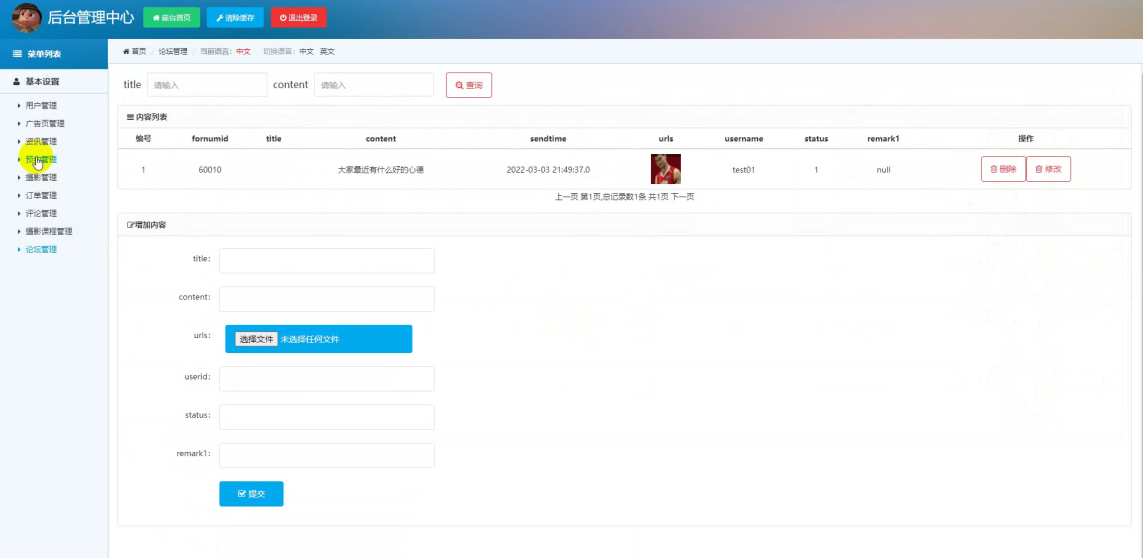


图5-7 课程管理模块测试图

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。