DAXUE2

**本科毕业论文（设计）**

**（2013届本科毕业生）**



**题　　目：** 基于Android的在线考试系统的设计与实现

**学生姓名：** 方明

**学生学号：** 13008038

**学院名称：** 科信软件学院

**专业名称：** 软件工程

**指导教师：** 张悦

**二零一七年六月**

**【摘要】**本系统是一个基于Android的在线考试系统。系统通过Java语言作为Android客户端的开发语言,通过PHP语言作为服务器端的开发语言,使用MySQL作为数据库服务器,通过Volley，使用JSON进行客户端和服务器数据交互。本系统主要分为学生模块和教师模块。学生模块主要包括注册、登录、答题、查分、选课以及修改个人信息。教师模块主要包括对试题库的增删改查、对学生成绩的查看以及修改个人信息。本系统可以帮助学生随时随地进行学习测试，提高教师的工作效率，真正实现随时随地进行考试操作。

**【关键词】**Android，Java，PHP，在线考试

Design and Implementation Online Examination System based on Android

**【Abstract】**This system is an online examination system based on Android. The system through the Java language as the Android client development, through the PHP language as a server-side development, using MySQL as the database server, client and server data to communicate via Volley by JSON. This system mainly divided into student and teacher module. Student module includes register, login, problem solving, check grade, course selection, and modify the personal information. Teacher's module mainly includes the question bank to add and delete, check the information of student achievement and modify the personal information. The system can help students to study and test anytime, anywhere to improve the efficiency of teachers, and truly achieve anytime, anywhere examination operation.

**【Keywords】**Android ,Java ,PHP,online examination system

目录

[1绪论 1](#_Toc480540244)

[1.1课题研究背景 1](#_Toc480540245)

[1.2 课题研究目的 1](#_Toc480540246)

[1.3 课题研究意义 1](#_Toc480540247)

[1.4 课题研究主要内容 1](#_Toc480540248)

[1.5 课题研究思路与研究方法 2](#_Toc480540249)

[2 相关技术介绍 3](#_Toc480540250)

[2.1 Android 3](#_Toc480540251)

[2.2 PHP 3](#_Toc480540252)

[2.3 数据库 4](#_Toc480540253)

[2.4 JSON 5](#_Toc480540254)

[2.5 Volley 5](#_Toc480540255)

[3 系统需求分析 6](#_Toc480540256)

[3.1 问题描述 6](#_Toc480540257)

[3.2 可行性分析 6](#_Toc480540258)

[3.3 功能需求分析 7](#_Toc480540259)

[4 系统的总体设计 9](#_Toc480540260)

[4.1 系统的总体设计 9](#_Toc480540261)

[4.2 客户端的总体设计 9](#_Toc480540262)

[4.3 服务端的总体设计 11](#_Toc480540263)

[4.4 数据库设计 13](#_Toc480540264)

[5 系统的详细设计 17](#_Toc480540265)

[5.1 模块实现 17](#_Toc480540266)

[5.2 学生模块 17](#_Toc480540267)

[5.3 教师模块 33](#_Toc480540268)

[5.4 工具类 41](#_Toc480540269)

[6 系统测试 46](#_Toc480540270)

[6.1 系统测试的目的 46](#_Toc480540271)

[6.2 系统测试的过程 46](#_Toc480540272)

[6.3 系统测试的功能点 46](#_Toc480540273)

[6.4 测试用例 46](#_Toc480540274)

[7 结论 57](#_Toc480540275)

[后记 58](#_Toc480540276)

[参考文献 59](#_Toc480540277)

1绪论

## 1.1课题研究背景

现如今，为了迎合数字化的网络环境，许多基于浏览器的在线考试系统投入使用。通过这种新的模式，学校形成了一种新的考试环境，提高了考试工作效率和标准化水平，使学校管理者、教师以及学生更加方便地通过网络端进行考试。但是，随之移动端的发展，基于浏览器的在线考试系统并不能解决当前发展的需求。因此，为了改进当前的考试模式，本文设计开发了基于Android的在线考试系统，真正实现了教师、学生随时随地通过移动终端进行考试操作。

## 1.2 课题研究目的

传统的考试需要进行考场编排、试卷打印以及试题批改，整个过程使得多次考试的投入成本较大。为了解决这个问题，实现无纸化、网络化、自动化的在线考试系统已经屡见不鲜。但是，当前的在线考试系统仍然基于浏览器实现，不免有一定的硬件条件以及网络的限制。而基于移动端的在线考试系统则突破了这些限制，并且Android平台兼容性高，设计一个基于Android的在线考试系统可以解决当前的问题，大大提高了考试效率。

## 1.3 课题研究意义

本系统可以帮助学生随时随地进行学习测试，提高教师的工作效率。本系统借助于Internet网络，使考试可以在手机等小型终端进行，也可以在异地进行，提供了在线考试、自动化测评、成绩查询、考题管理等功能。这种无纸化的即时在线考试系统，使得传统考试突破时空限制，提高了一场考试的效率，减少了一场考试的投资，真正实现随时随地进行考试操作。

## 1.4 课题研究主要内容

开发一个基于Android的在线考试系统。Android端提供在线考试、自动化测评、成绩查询、考题管理等功能。服务端采用PHP的ThinkPHP的MVC框架进行开发，连接MySQL数据库，为Android端提供一系列数据交互URL地址。本论文主要介绍了对基于Android的在线考试系统的分析，以及设计开发的全部过程。本课题采用软件工程开发的思想，对系统结构进行开发。

## 1.5 课题研究思路与研究方法

本系统是基于Android端的在线考试系统，最重要的难题是解决Android端与服务端的数据交互。在进行现有的技术分析后，本系统采用Android提供的Volley的jar包进行数据交互，同时采用JSON作为数据交互的类型。同时，为了使本系统的可扩展性和可维护性，本系统最终采取MVC架构进行开发。Android端综合利用了Activity、Intent、Service等核心组件来完成简单的逻辑，服务端则利用ThinkPHP框架完成数据处理，与Android端完成HTTP的请求和响应。

2 相关技术介绍

## 2.1 Android

2.1.1 Android概述

Android是一个基于Linux平台的开源手机操作系统，是目前全球最受欢迎的移动平台。Android在全球190多个国家拥有数亿个移动设备。 它是任何移动平台最大的安装基础，并且日益增长，每天还有一百万用户首次启动他们的Android设备，并开始寻找应用程序，游戏和其他数字内容。Android不断推出硬件和软件的边界，为用户和开发人员带来新功能。 Android技术的快速发展使开发的应用程序能够在强大的差异化应用程序中保持领先地位。为了提高开发的效率，Android Developer Tools提供了一个完整的Java IDE，具有用于开发，调试和打包Android应用程序的高级功能。使用IDE，可以在任何可用的Android设备上进行开发，也可以创建仿真任何硬件配置的虚拟设备。

Android 提供了一个自适应应用框架，可用以为不同的设备配置提供独特的资源。例如，您可以针对不同的屏幕尺寸创建不同的 XML 布局文件，系统将根据当前设备的屏幕尺寸确定要应用的布局。

2.1.2 Android的优势

Android作为目前用户最多的移动平台，具备了先天的优势。Android提供了一个世界一流的平台，为各地的Android用户创建应用和游戏，以及一个开放的市场，立即分发给他们。Android平台手机也能够使用很多Google提供的优秀的API。目前Android平台手机在学生中占有相当大的比例，在前期系统使用或推广上具有其他平台所不具备的优势。Android的开源免费也使本系统的开发更为方便灵活。

2.1.3 Android Studio

Android Studio是一个视窗化的开发环境。Android Stuido是Google推出，专门为Android“量身订做”的，是Google大力支持的一款基于IntelliJ idea改造的IDE。Android Studio提供了Android专属的重构和快速修复，能够基于模板的向导来生成常用的Android应用设计和组件。Studio亮相之处就支持Gradle，可以说Gradle集合了Ant和Maven的优点，不管是配置、编译、打包都非常棒。

## 2.2 PHP

2.2.1 PHP概述

PHP是一种通用的开源脚本语言，是目前最流行的WEB开发语言。PHP 是一种 HTML 内嵌式的语言，是一种在服务器端执行的嵌入HTML文档的脚本语言，语言的风格有类似于C语言，被广泛的运用。PHP还支持几乎所有流行的数据库以及操作系统。PHP的数据类型是弱类型，使PHP的开发更为灵活，也使PHP更易学习。

2.2.2 PHP的优势

作为Android应用的服务端，PHP的灵活性以及快捷性正好符合了当前的需求。PHP较与JAVA，PHP更加适合快速开发的环境。PHP也能够提供JSON数据的编译方法，使服务端与Android端数据交互更为方便。目前PHP也有许多优秀的开源框架，如ThinkPHP、Zend Framework、Laravel等

2.2.3 ThinkPHP

ThinkPHP是一个快速、简单的基于MVC和面向对象的轻量级PHP开发框架，倡导大道至简，开发由我的开发理念，用最少的代码完成更多的功能。其中提供的路由功能是针对模块设置的，因此使得PHP提供的数据交互URL地址更加规范。ThinkPHP自动生成MVC架构文件夹也适合新接触MVC架构的初学者使用，使服务端的开发更为规范，使未来的维护更为方便，也符合JAVA的面向对象设计的特性。

2.2.4 Wamp Server

Wamp Server是一个基于windows操作系统，是Apache、MySQL、PHP的一个集成环境，快速地为本系统搭建了一个安全高效的WEB服务器。Wamp Server简洁的操作界面省去了修改配置文件的麻烦，使服务端的URL传输数据测试更加稳定更加快捷。

## 2.3 数据库

2.3.1 MySQL

本系统采用了MySQL作为服务端的数据库语言。MySQL是目前与PHP配套使用的最流行的开源数据库系统。不管在小型还是大型的应用程序中，MySQL都是理想的选择。MySQL非常快速，可靠，且易于使用。同时也支持标准的SQL。MySQL对PHP有很好的支持，搭配PHP以及Apache组成一个良好的开发环境。

2.3.2 SQLite

SQLite 是一款非常流行的嵌入式数据库，它支持 SQL 查询，并且只用很少的内存。Android 在运行时集成了 SQLite，所以每个 Android 应用程序都可以使用 SQLite 数据库。对于熟悉 SQL 的开发人员来说，使用 SQLite 相当简单。较于JDBC这种需要消耗太多内存的API，SQLite更适合Android开发。

## 2.4 JSON

2.4.1 JSON概述

JSON是一种轻量级的数据交换格式。JSON是一种完全与语言无关的文本格式，JSON解析器和JSON库支持许多不同的编程语言，JSON还具有自我描述性，更易理解。这些属性使JSON成为理想的数据交换语言。

2.4.2 JSON的优势

相比于xml这种数据交换格式来说，因为解析xml比较的复杂，而且需要编写大段的代码，所以客户端和服务器的数据交换格式往往通过JSON来进行交换。Android也提供了比较成熟的JSON第三方Jar包，而PHP对JSON数据的解析有封装好的方法。本系统需要实现的功能较小，JSON的数据传输方式满足于当前的需求。而JSON的自我描述性，是本系统的测试更为方便直观。

## 2.5 Volley

Android系统中主要提供了两种方式来进行HTTP通信，HttpURLConnection和HttpClient，几乎在任何项目的代码中我们都能看到这两个类的身影，使用率非常高。不过HttpURLConnection和HttpClient的用法还是稍微有些复杂的，如果不进行适当封装的话，很容易就会写出不少重复代码。于是乎，一些Android网络通信框架也就应运而生，比如说AsyncHttpClient和Universal-Image-Loader。Volley可是说是把AsyncHttpClient和Universal-Image-Loader的优点集于了一身，既可以像AsyncHttpClient一样非常简单地进行HTTP通信，也可以像Universal-Image-Loader一样轻松加载网络上的图片。除了简单易用之外，Volley在性能方面也进行了大幅度的调整，它的设计目标就是非常适合去进行数据量不大，但通信频繁的网络操作。

3 系统需求分析

## 3.1 问题描述

基于Android的在线考试系统为了给教师、学生提供一个良好的考试环境。学生可以在该系统中先进行选择需要考试的课程，系统则从服务端下载相应的试卷提供给学生进行考试，学生答题后，该课程的试题将不再显示，学生可从成绩查询一栏查询自己的成绩。教师可以在该系统中修改自己负责的课程的试题，增加试题以及删除试题，教师也可从查询成绩一栏查看当前课程所有学生的成绩。该系统充分利用了网络技术，实现无纸化考试。系统模块主要分为学生模块和教师模块。学生模块主要包括注册、登录、答题、查分、选课以及修改个人信息。教师模块主要包括对试题库的增删改查、对学生成绩的查看以及修改个人信息。

## 3.2 可行性分析

传统考试要求出题、打印试题、安排考场、试卷批阅等一系列流程，效率低而且成本高。因此，对现有的常规考试模式进行改进，设计一个方便高效的考试方式显得尤为重要。近年来，虽然基于浏览器的考试系统屡见不鲜，但是随着无线网络以及智能手机的发展，设计一款基于Android的在线考试系统更为符合目前的需求。

3.2.1 技术可行性

本系统主要使用Android Studio开发工具进行开发，Android Studio作为Android官方的IDE，为Android开发提供了一系列便利的API，使得开发更为快速完整。PHP作为新起的热门语言，其灵活性级快捷性使其作为服务端的开发语言。JSON提供的数据传输方式，使客户端与服务端的数据交互更为便利。而作为服务端的数据库语言MySQL，不管在小型还是大型的应用程序中，MySQL都能够高效安全地存储数据。SQLite则轻便快捷，在Android领域被广泛应用，采用SQLite可以为本系统Android端提供暂时存储数据的服务。

3.2.2 经济可行性

本系统均由开源平台进行开发，功能较小，减少了开发费用。本系统也为了节约教师以及学生的时间，减少了人力开销，以及考试的纸张成本。

3.2.3 社会因素方面可行性

本系统开发均用正版的软件。系统的页面简洁明了，操作简单，能够满足目前的需求。

## 3.3 功能需求分析

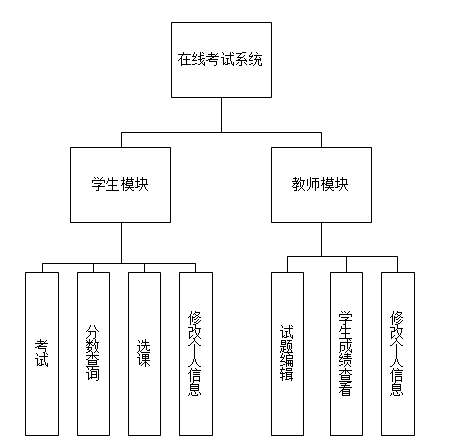
为了实现一个功能完整的在线考试系统，这个系统的功能模块如下：

图3-1 功能模块图

3.3.1 学生模块

实现了学生答题、分数查询、选课、修改个人信息功能。学生需要先在选课功能下选择需要进行考试的科目，返回主菜单后进入答题，系统则从服务器下载对应课程的试题，学生答完所有试题并确认提交后，该课程的试题将不显示，学生可从分数查询功能下查询自己当前答完试题的成绩。学生可在修改个人信息功能下对个人信息进行更改，为了保证学生信息的完整性，该系统只允许学生修改数据库中存储的手机号码以及密码。

3.3.2 教师模块

实现了教师试题编辑、查看学生成绩、修改个人信息功能。教师只能修改自己所管理的课程的所有试题，对当前已有的试题进行修改、删除，也可以添加新的试题，操作完成后，学生端该课程的试题目录将更新。教师可以通过学生成绩查看功能查看当前课程所有学生的成绩，通过这个功能对学生成绩进行统计。教师也可在修改个人信息功能下对个人信息进行更改，为了保证教师信息的完整性，该系统只允许教师修改数据库中存储的手机号码以及密码。

4 系统的总体设计

## 4.1 系统的总体设计

基于Android的在线考试系统分为服务端和客户端，采用C/S架构。服务端的数据库为主数据库，存储用户信息、课程信息、试题信息以及之间的联系。客户端以Android平台进行开发，以SQLite为暂存数据库，存储临时性信息。

本系统的服务端和客户端均采用MVC的设计模式。服务端采用ThinkPHP框架，为客户端提供数据传输API。

## 4.2 客户端的总体设计

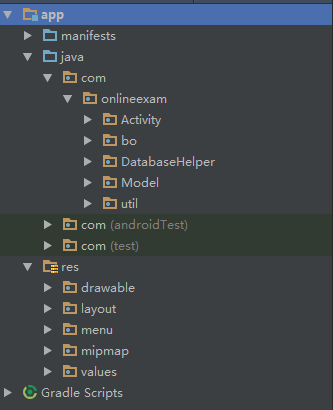
客户端的架构分为Activity、bo、DatabaseHelper、Model和util，用来处理Android的代码逻辑，而Android的页面编写的xml文件存放在layout中。具体架构如下：

图4-1 客户端架构

1. Activity包

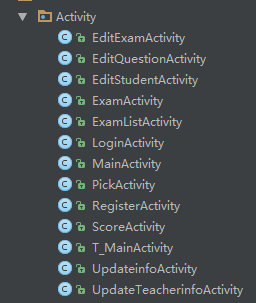
Activity作为Android的一大核心组件，负责实现用户界面，处理系统交互动作。具体内容如下：

图4-2 Activity包

1. bo包

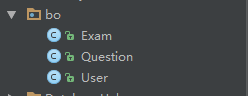
bo是数据对象的一个package，本系统的主要数据对象分为用户、课程和题目。因此，本系统的bo对象以user、course、question表进行创建。具体如下：

图4-3 bo包

1. DatabaseHelper包

DatabaseHelper包是对SQLite数据库的创建，库中表的创建和删除操作。具体如下：

图4-4 DatabaseHelper包

1. Model包

Model包是数据交互层存储的包，主要是本系统对SQLite数据库的增删改查的操作，提高系统的可维护性和可扩展性。由于本系统的主要bo对象为用户、课程和题目，因此Model对象以对这三个对象进行操作。具体如下：

图4-5 Model包

1. util包

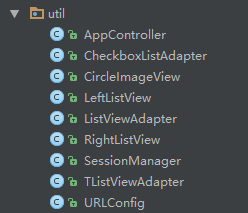
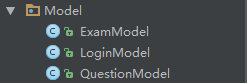
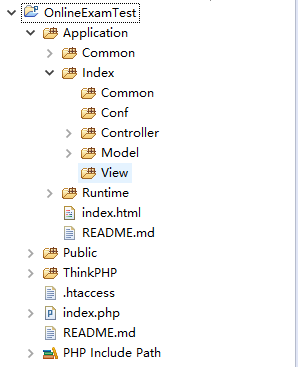
util为工具类存储的包。封装了本系统的适配器类、Session控制类、以及URL的配置类。具体如下：

图4-6 util包

## 4.3 服务端的总体设计

服务端使用的是ThinkPHP提供的MVC框架，服务端没有具体的View页面，只为客户端提供数据交互的API。具体架构如下：

图4-7 服务端架构

1. Controller包

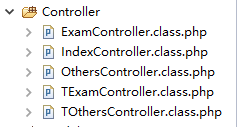
Controller包主要以客户端的功能模块进行划分，为客户端提供一个直观的API。具体如下：

图4-8 Controller包

1. Model包

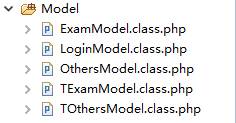
Model包主要是服务端与MySQL数据库交互的数据交互层。增加服务端的可维护性和可扩展性。具体如下：

图4-9 Model包

## 4.4 数据库设计

该系统主要采用开源的MySQL数据库，Android端采用SQLite数据库

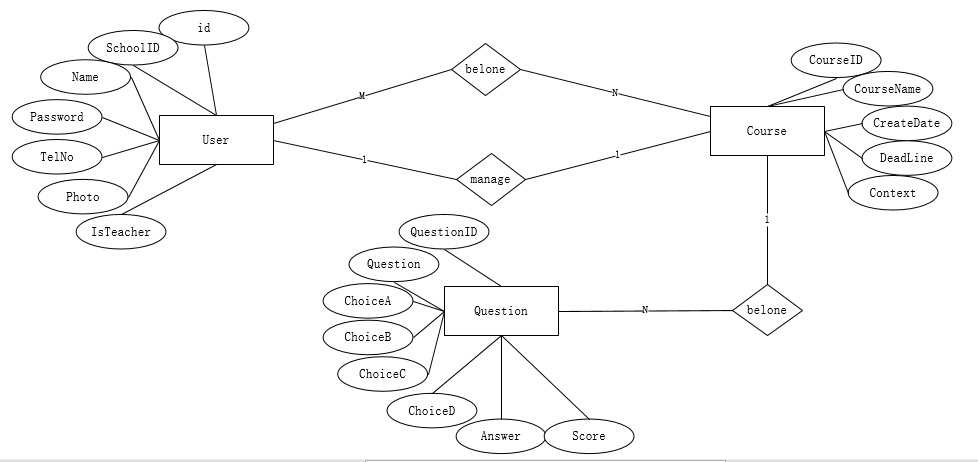
4.4.1 系统E-R图

图4-10 系统E-R图

4.4.2 服务端数据库表设计

MySQL非常快速，可靠，且易于使用。同时也支持标准的SQL。MySQL对PHP有很好的支持。

1. 表4-1 为所有用户表（user）：

表4-1 user表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| id | int | 主键，自增 | 是 | 非空 |
| SchoolID | int | 学号 |  | 非空 |
| Name | varchar | 姓名 |  | 非空 |
| Password | varchar | 密码 |  | 非空 |
| TelNo | varchar | 联系方式 |  | 非空 |
| Photo | varchar | 默认空字符串 |  | 非空 |
| IsTeacher | int | 是否为教师，默认值为0 |  | 非空 |

用户表存储了学生以及教师的基本信息，由IsTeacher字段进行判定该信息是否为教师。Photo字段存储的为该用户的头像地址，若为空，默认读取Android图片。

1. 表4-2 课程表（course）：

表4-2 course表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| CourseID | int | 主键，自增 | 是 | 非空 |
| CourseName | varchar | 课程名称 |  | 非空 |
| CreateDate | date | 课程创建时间 |  | 非空 |
| DeadLine | date | 课程提交时间 |  | 非空 |
| Context | varchar | 课程说明 |  | 非空 |

课程表存储了基本的课程信息。

1. 表4-3试题表（question）：

表4-3 question表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| QuestionID | int | 主键，自增 | 是 | 非空 |
| Question | varchar | 试题题目 |  | 非空 |
| ChoiceA | varchar | 选项A |  | 非空 |
| ChoiceB | varchar | 选项B |  | 非空 |
| ChoiceC | varchar | 选项C |  | 非空 |
| ChoiceD | varchar | 选项D |  | 非空 |
| Answer | varchar | 试题答案 |  | 非空 |
| Score | int | 试题分数 |  | 非空 |

试题表存储了所有试题信息。

1. 表4-4学生-课程表（course\_student）：

表4-4 course\_student表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| CourseID | int | 课程编号，外键 | 是 | 非空 |
| StudentID | int | 学生编号，外键 | 是 | 非空 |
| Score | int | 学生成绩 |  | 非空 |
| IsAnswered | int | 是否作答 |  | 非空 |

学生-课程表存储了学生选择课程的信息，以及该课程作答后的分数。

1. 表4-5教师-课程表（course\_teacher）：

表4-5 course\_teacher表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| CourseID | int | 课程编号，外键 | 是 | 非空 |
| TeacherID | int | 教师编号，外键 | 是 | 非空 |

教师-课程表存储了课程所对应的教师的编号。

1. 表4-6课程-试题表（course\_question）：

表4-6 course\_question表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| CourseID | int | 课程编号，外键 | 是 | 非空 |
| QuestionID | int | 试题编号，外键 | 是 | 非空 |

课程-试题表存储了课程所对应的所有试题。

4.4.3 Android端数据库表设计

SQLite 是一款非常流行的嵌入式数据库，在Android中的应用极为广泛。为本系统Android端暂时存储数据的功能，建立了服务端主要的三个数据库表。

1. 表4-7 所有用户表（user）：

表4-7 user表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| id | Integer | 主键，自增 | 是 | 非空 |
| SchoolID | Integer | 学号 |  | 非空 |
| Name | Text | 姓名 |  | 非空 |
| Password | Text | 密码 |  | 非空 |
| TelNo | Text | 联系方式 |  | 非空 |
| Photo | Text | 默认空字符串 |  | 非空 |
| IsTeacher | Integer | 是否为教师，默认值为0 |  | 非空 |

用户表存储了学生以及教师的基本信息，由IsTeacher字段进行判定该信息是否为教师。Photo字段存储的为该用户的头像地址，若为空，默认读取Android图片。

1. 表4-8 课程表（course）：

表4-8 course表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| CourseID | Integer | 主键，自增 | 是 | 非空 |
| CourseName | Text | 课程名称 |  | 非空 |
| CreateDate | Text | 课程创建时间 |  | 非空 |
| DeadLine | Text | 课程提交时间 |  | 非空 |
| Context | Text | 课程说明 |  | 非空 |

课程表存储了基本的课程信息。

1. 表4-9试题表（question）：

表4-9 question表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 说明 | 主键 | 非空 |
| QuestionID | Integer | 主键，自增 | 是 | 非空 |
| CourseID | Integer | 外键，课程编号 |  | 非空 |
| Question | Text | 试题题目 |  | 非空 |
| ChoiceA | Text | 选项A |  | 非空 |
| ChoiceB | Text | 选项B |  | 非空 |
| ChoiceC | Text | 选项C |  | 非空 |
| ChoiceD | Text | 选项D |  | 非空 |
| Answer | Text | 试题答案 |  | 非空 |
| Score | Integer | 试题分数 |  | 非空 |

试题表存储了所有试题信息。

5 系统的详细设计

## 5.1 模块实现

本系统主要需要实现学生模块和教师模块。在登录界面，用户输入自己的学号和密码，系统对该用户的学生或者教师身份进行判断，登录成功后学生进入学生界面，教师进入教师界面。在学生界面中，学生可以从程序的侧边栏选择进行考试、查分、选课以及修改个人信息。在教师界面中，教师可以从程序的侧边栏选择进行试题的操作、查询学生分数以及修改个人信息。

## 5.2 学生模块

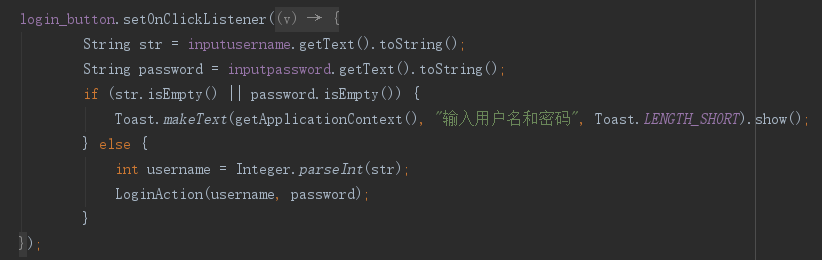
5.2.1 登录功能

学生与教师共享同一个登录界面，登录界面包括了用户ID和密码两个输入框，以及一个跳向注册界面的链接。界面如图5-1：

图5-1 登录界面



学生在登录操作的时候，Activity会进行输入框的空值判断，若用户ID或者密码没有输入，则弹窗显示“输入用户名和密码”的信息。具体代码如下：

图5-2 空值判断代码

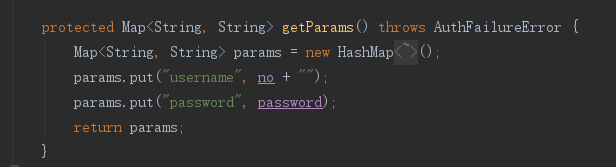
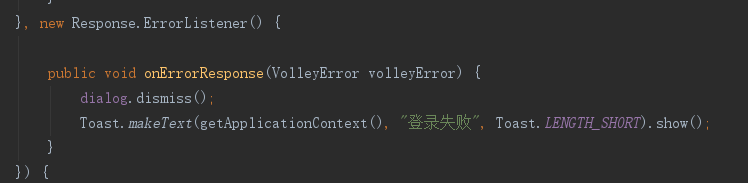
输入完整的用户ID和密码后点击登录，LoginActivity则调用了StringRequest类向服务端传输数据并接收数据。具体代码如下：

图5-3 传输数据代码

图5-4 接收数据成功代码

图5-5 接收数据失败代码

将数据封装在StringRequest这个对象中，调用Volley的RequestQueue方法向服务端进行信息交互。客户端访问服务端提供的URL（etc.http://192.168.1.102/onlineexamtest/index/index/login），传输方式为POST。最后接收数据为JSON，具体如下：

表5-1 登陆JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 登录失败 | {  "error": true,  "errormsg": "向服务器提交失败"  } |
| 登录失败 | {  "error": true,  "errormsg": "密码错误"  } |
| 登录失败 | {  "error": true,  "errormsg": "不存在该账户"  } |
| 登录成功 | {  "error": false,  "userinfo": {  "id": "1",  "SchoolID": "13008038",  "Name": "方明",  "Password": "123456",  "TelNo": "18502410459",  "Photo": "",  "IsTeacher": "0"  }  } |

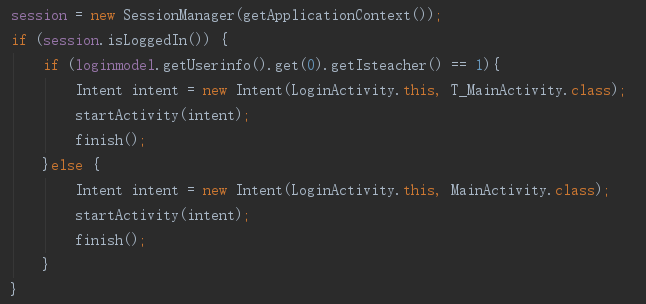
学生登录成功后，进入学生主界面。同时，系统会创建session记录登录状态，登录信息则存储在Android机的SQLite数据库中。在每一次进入本系统时，系统都会判断一次登录状态，如果是未登录的用户，则进入登录界面，如果是已登录的用户，则直接进入主界面进行操作。部分session代码如下：

图5-6 session代码

5.2.2 注册功能

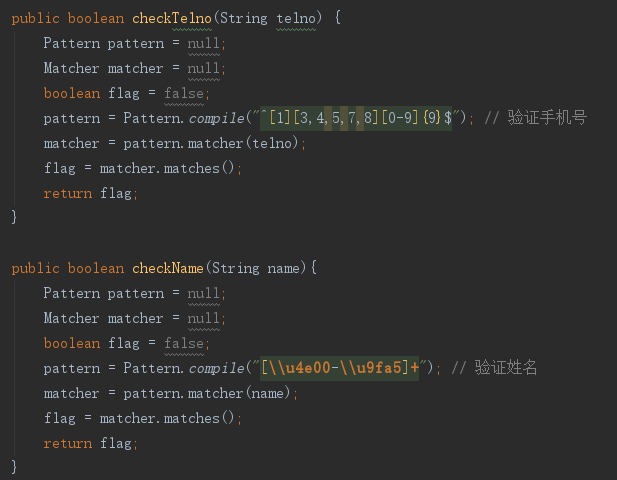
注册功能是学生模块独有的功能，通过注册生成的用户的身份均为学生。在注册界面需要输入用户ID、姓名、密码、确认密码、电话信息进行注册。同时注册界面也有跳回登录界面的链接。界面如图5-7：

图5-7 注册界面



在RegisterActivity中，系统会先对输入框进行检查。用户名ID、密码不能为空，确认密码需与密码相同，姓名用正则验证中文，电话信息用正则验证11位正确密码。具体代码如图5-8、图5-9：

图5-8 注册验证

图5-9 手机号、姓名正则验证

客户端将用户ID、密码、姓名、手机号对应存储为SchoolID、Password、Telno、Name用Post的方式传输给服务端。服务端返回的JSON格式如下：

表5-2 注册JSON格式

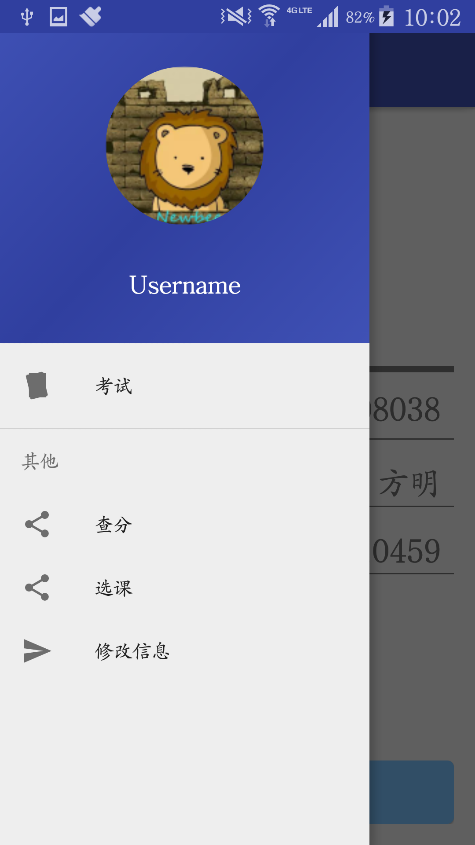
|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 注册失败 | {  "error": true,  "errormsg": "向服务器提交失败"  } |
| 注册失败 | {  "error": true,  "errormsg": "已存在该账户"  } |
| 注册失败 | {  "error": true,  "errormsg": "注册失败"  } |
| 注册成功 | {  "error": false  } |

注册成功后系统将返回登录界面，提供给用户登录。

5.2.3 主界面功能

用户主界面显示用户的头像、用户学号、姓名、电话、以及一个退出登录按钮。主界面还包括一个侧滑栏，用户可以通过侧滑栏进入考试、查分、选课、修改信息等功能。主界面如图5-10、图5-11：

图5-10 主界面

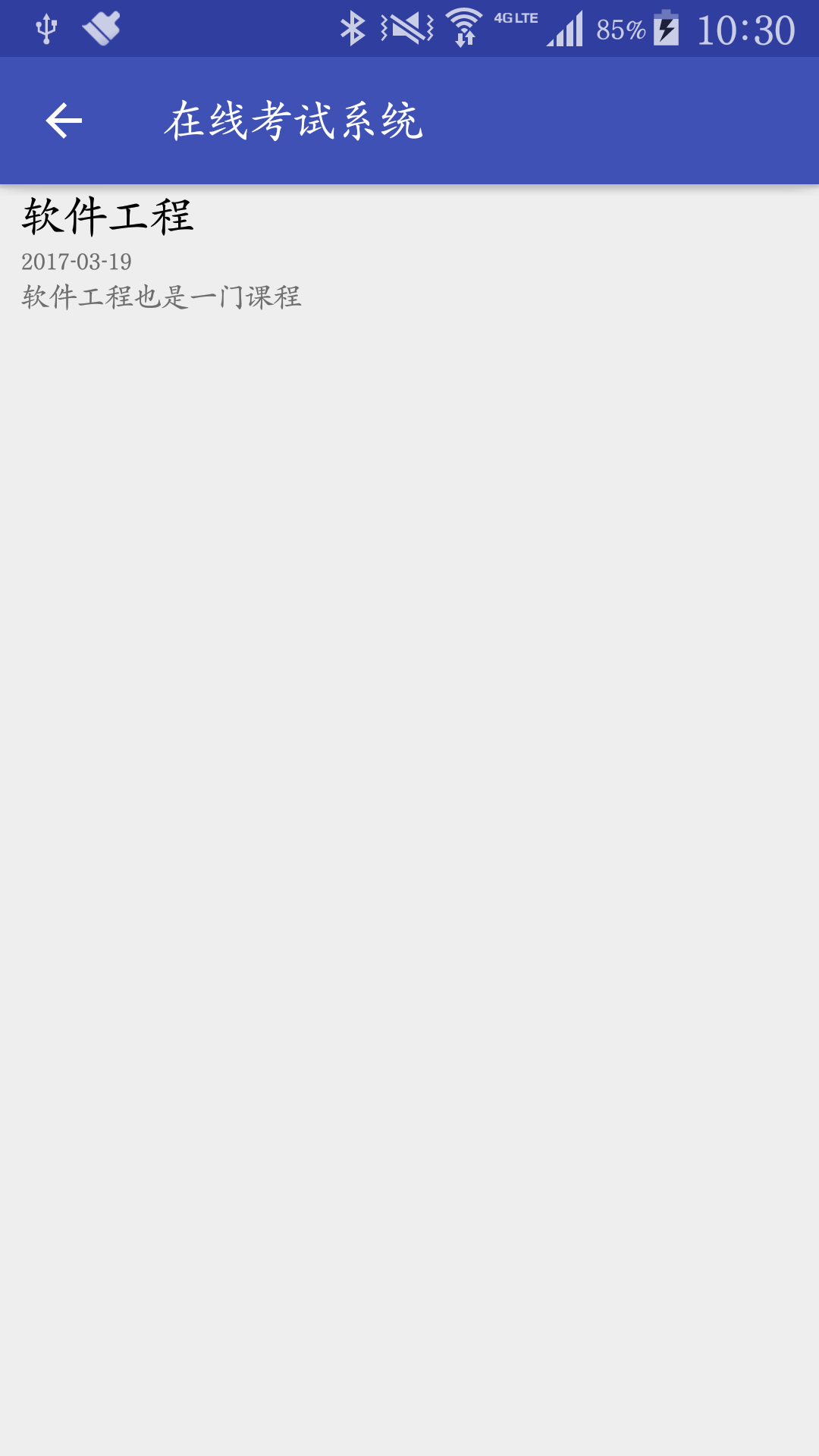
图5-11 主界面侧滑栏

主界面的信息是由Android机所存储的user表中获得，点击退出后，本系统将session的登录状态设为false，并将user表中的数据清除。点击退出后，系统将返回登录界面，下一回进入本系统时，将不再进入主界面，需要用户登录后才允许进入主界面。这个功能保证了账户的安全，也解决了同一账户多次登录的麻烦。

5.2.4 考试功能

学生点击考试进入考试界面，向服务端获取相应的试题。考试界面如图5-12：

图5-12 考试界面



每个ListView都有一个侧滑功能，可以查看详情或者直接进入答题。如图5-13、图5-14：

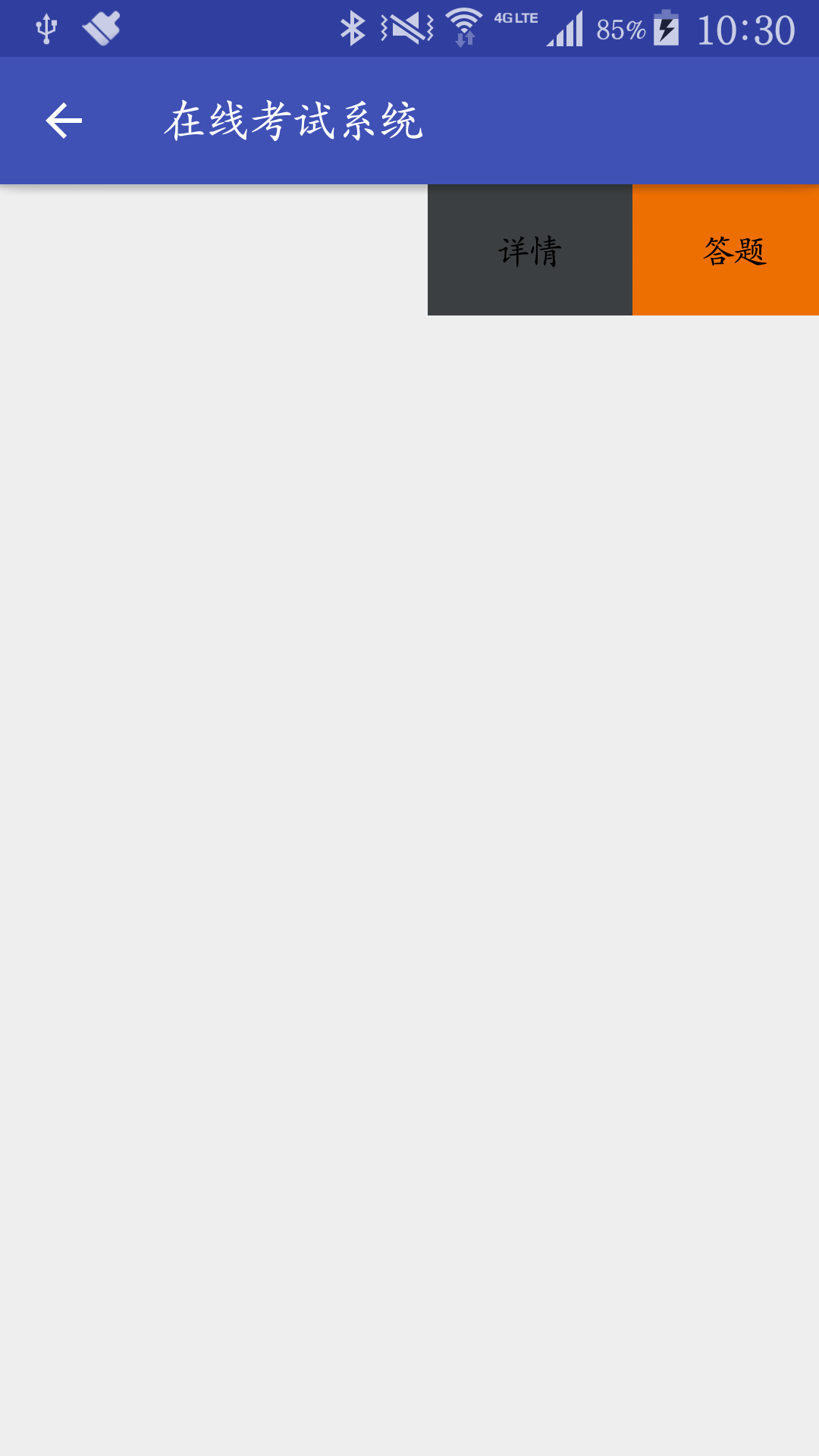


图5-13 侧滑功能

图5-14 详情功能



服务端传输给客户端的JSON格式如下：

表5-3 获取试题JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 获取试题 | [  {  content=软件工程也是一门课程,  time=2017-03-19,  CourseID=2,  title=软件工程  }  ] |
| 没有试题 | {  "error": true,  "errormsg": "当前没有课程"  } |
| 获取失败 | {  "error": true,  "errormsg": "向服务器传输数据失败"  } |

进入答题功能，具体界面如图5-15、图5-16：

图5-15 答题界面



图5-16 答题界面

系统根据获得试题的数量生成对应的fregment，用户进行切换为每个试题进行作答，当用户滑动到最后一个试题后，系统将提示是否提交试卷，取消将可以继续作答，确定将提交试卷，系统进行自动判断生成分数。作答完成后，该套试题将不再显示给该用户，用户通过查询成绩功能查看当前已完成试题的分数。

获取试题的JSON格式如下：

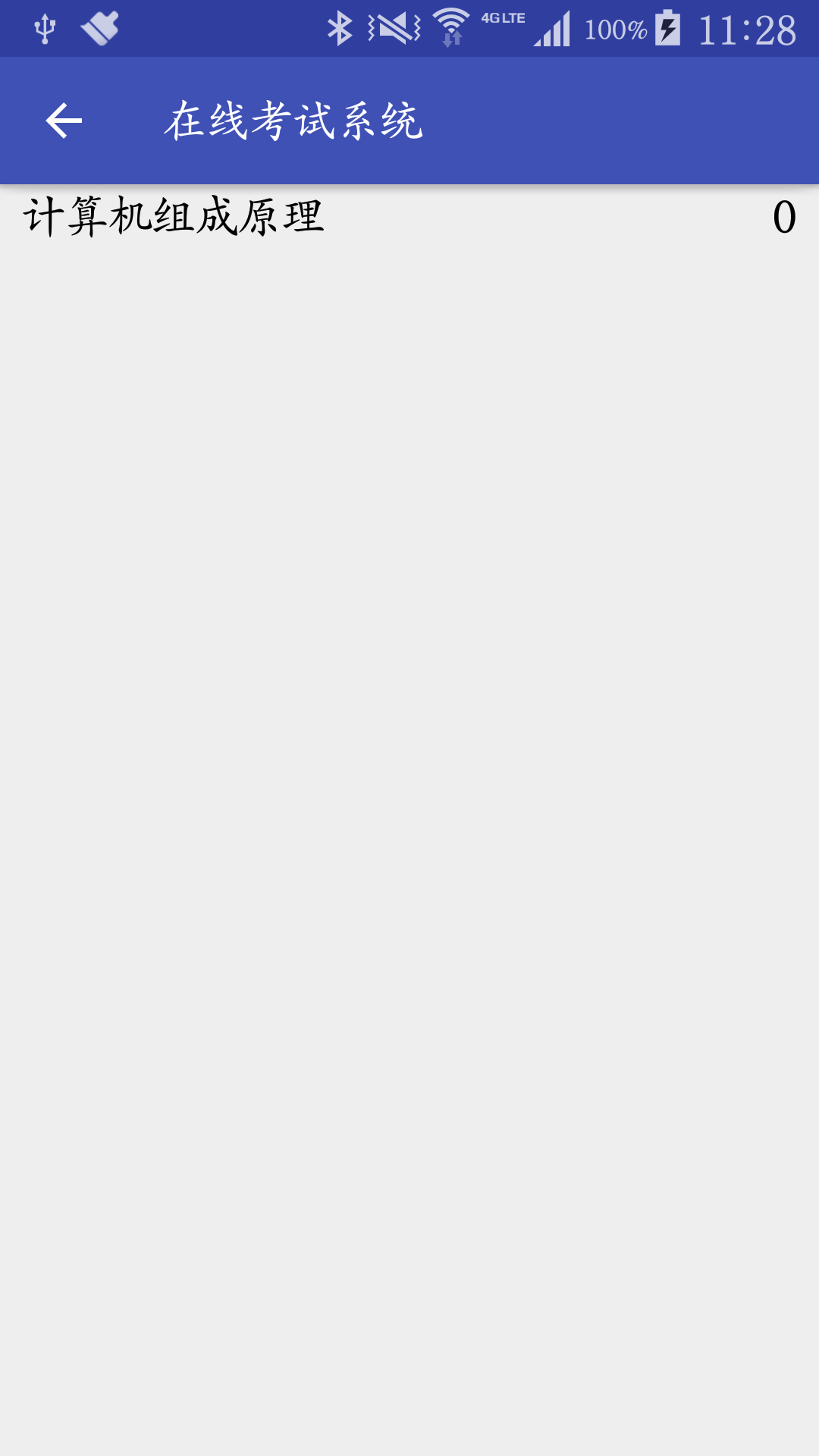
表5-4 获取试题JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 获取成功 | {  "error": false,  "questionlist": [  {  "QuestionID": "1",  "Question": "软件工程怎么样",  "ChoiceA": "A.特别棒",  "ChoiceB": "B.不错",  "ChoiceC": "C.一般般",  "ChoiceD": "D.不喜欢",  "Answer": "A",  "Score": "10"  },  {  "QuestionID": "2",  "Question": "软件工程怎么样",  "ChoiceA": "A.特别棒",  "ChoiceB": "B.不错",  "ChoiceC": "C.一般般",  "ChoiceD": "D.不喜欢",  "Answer": "B",  "Score": "10"  },  {  "QuestionID": "3",  "Question": "软件工程怎么样",  "ChoiceA": "A.特别棒",  "ChoiceB": "B.不错",  "ChoiceC": "C.一般般",  "ChoiceD": "D.不喜欢",  "Answer": "C",  "Score": "10"  },  {  "QuestionID": "4",  "Question": "软件工程怎么样",  "ChoiceA": "A.特别棒",  "ChoiceB": "B.不错",  "ChoiceC": "C.一般般",  "ChoiceD": "D.不喜欢",  "Answer": "D",  "Score": "10"  }  ]  } |
| 没有试题 | {  "error": true,  "errormsg": "当前没有试题"  } |
| 获取失败 | {  "error": true,  "errormsg": "向服务器传输数据失败"  } |

5.2.5 查分功能

用户返回主界面点击查分按钮进入查分界面，查分界面显示所有当前已完成试题的分数。查分界面如图5-17：

图5-17 查分界面



查分功能从服务端获取的JSON格式如下：

表5-5 获取成绩JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 获取成绩 | {  "error": false,  "scorelist": [  {  "CourseID": "1",  "CourseName": "计算机组成原理",  "CreateDate": "2017-03-19 21:13:15",  "DeadLine": "2017-03-20 21:13:23",  "Context": "计算机组成原理是一门课程",  "Score": "0"  }  ]  } |
| 没有成绩 | {  "error": true,  "errormsg": "当前没有成绩"  } |
| 获取失败 | {  "error": true,  "errormsg": "向服务器传输数据失败"  } |

5.2.6 选课功能

用户返回主界面点击选课按钮进入选课界面，选课界面显示所有当前用户未选课程。选课界面如图5-18：

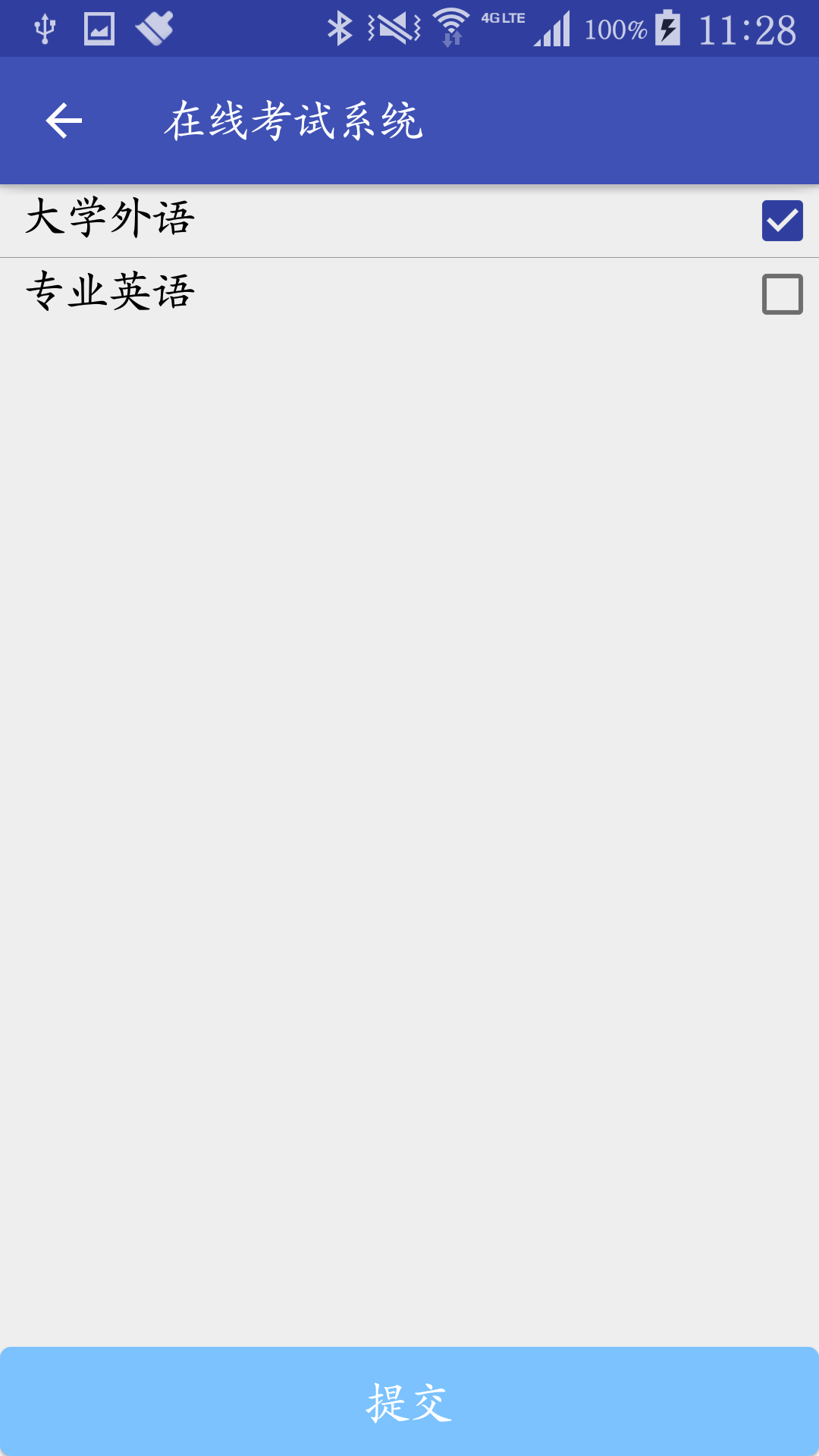


图5-18 选课界面

点击提交后，系统向服务端传输已选的课程，服务端返回的JSON格式如下：

表5-6 选课JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 选课成功 | {  "error": false  } |
| 选课失败 | {  "error": true,  "errormsg": "选课失败"  } |
| 提交失败 | {  "error": true,  "errormsg": "向服务器传输数据失败"  } |

5.2.7 修改信息功能

用户返回主界面点击修改信息按钮进入修改信息界面，修改信息界面显示当前用户的信息，为了保证用户信息的完整性，本系统仅允许用户修改电话和密码。用户点击提交后，系统会优先判断用户是否修改了成绩，避免不必要的提交，同时系统也会对输入的手机号码进行正则匹配，通过验证后向服务器提交。修改信息界面如图5-19：



图5-19 修改信息界面

服务端信息处理后返回的JSON格式如下：

表5-7 修改信息JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 更改成功 | {  "error": false  } |
| 更改失败 | {  "error": true,  "errormsg": "向服务器更新失败"  } |
| 提交失败 | {  "error": true,  "errormsg": "向服务器传输数据失败"  } |

## 5.3 教师模块

5.3.1 登录功能

教师与学生共享同一个登录界面，登录界面见图5-1，系统对传输回来的IsTeacher字段进行判断，登录成功后进入教师主页面。教师身份的账号不能通过注册生成，需要通过后台管理员赋予权利。

服务端传输的JSON格式如下：

表5-8 登录JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 登录失败 | {  "error": true,  "errormsg": "向服务器提交失败"  } |
| 登录失败 | {  "error": true,  "errormsg": "密码错误"  } |
| 登录失败 | {  "error": true,  "errormsg": "不存在该账户"  } |
| 登录成功 | {  "error": false,  "userinfo": {  "id": "2",  "SchoolID": "13000000",  "Name": "方明",  "Password": "456789",  "TelNo": "18502410459",  "Photo": "",  "IsTeacher": "1"  }  } |

5.3.2 主界面功能

教师通过登录验证后进入主界面，主界面显示该教师的基本信息。教师可以通过主界面的侧滑栏选择相应的功能进入相应的界面，教师拥有的功能有修改题库、查看学生成绩、修改信息。教师登录成功后，系统会用session修改登录状态，下次进入本系统则直接进入主界面，同时也会通过Android机存储的SQLite数据库的user表检测该用户的类型进入相应的主界面。教师点击退出后，session将登录状态改为false，并清空Android机数据库中的user表。教师的主界面功能如图5-20、图5-21：

图5-20 教师主界面

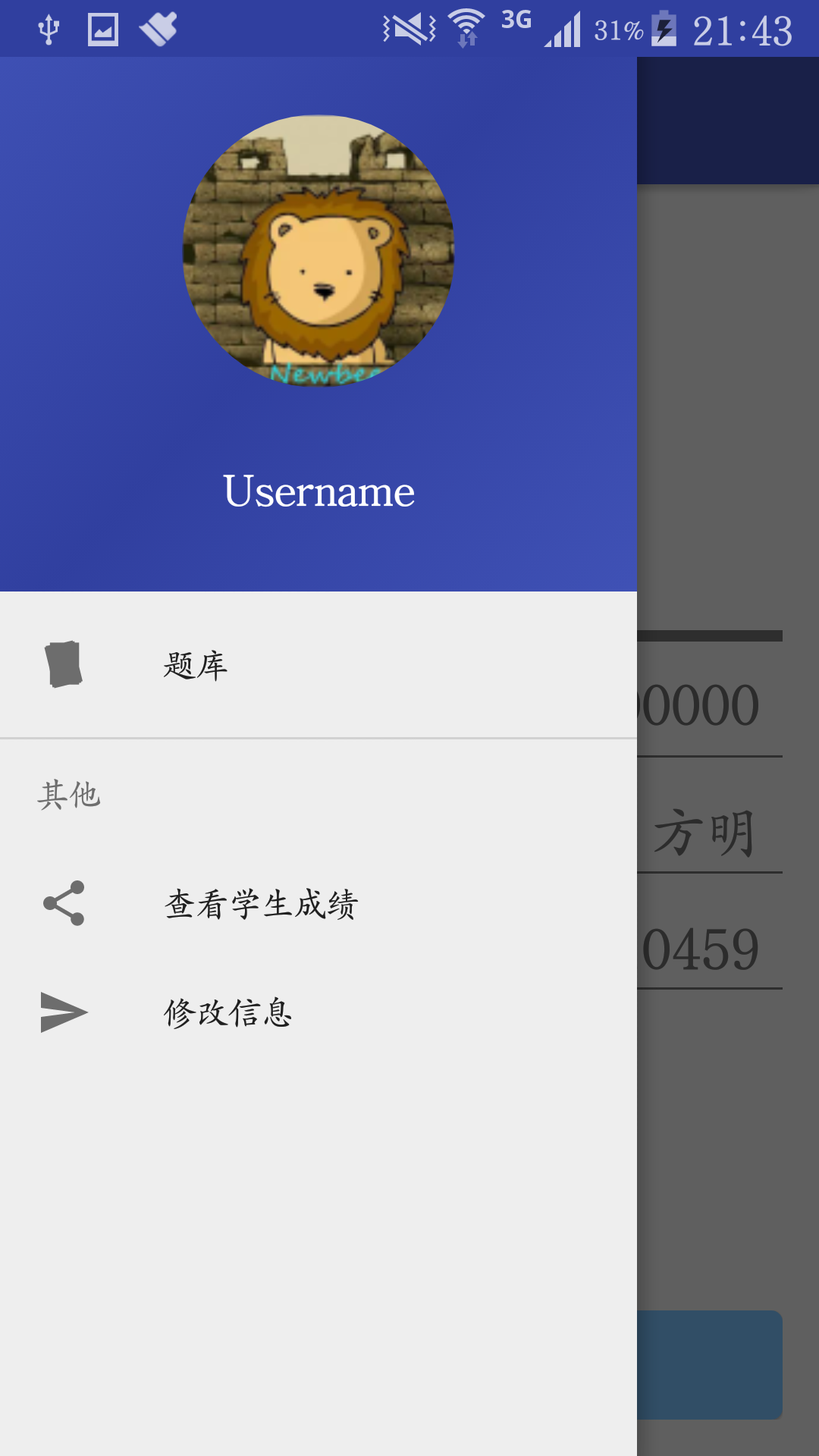


图5-21 教师主界面侧滑栏

5.3.3 题库功能

教师通过主界面的侧滑栏菜单，进入题库界面，题库显示了教师所管理的课程的所有试题，教师可以在试题界面增加新试题，删除已有的试题。增加一个试题，试题内容均有默认值，试题分数默认为零。删除的试题没有办法再恢复。教师可以通过每道试题的侧滑栏进入试题编辑功能。题库界面如图5-22：



图5-22 题库功能

教师点击界面右下角悬浮按钮可添加一条试题数据，如图5-23：

图5-23 新建试题功能



教师左滑试题点击删除即可删除一条试题，如图5-24、图5-25：

图5-24 左滑菜单功能



图5-25 删除试题

教师点击左滑菜单修改按钮，进入修改试题界面，修改完成后点击提交，弹窗输入该试题的分数，系统将修改完成后的试题上传到服务器的数据库中。如图5-26、图5-27：

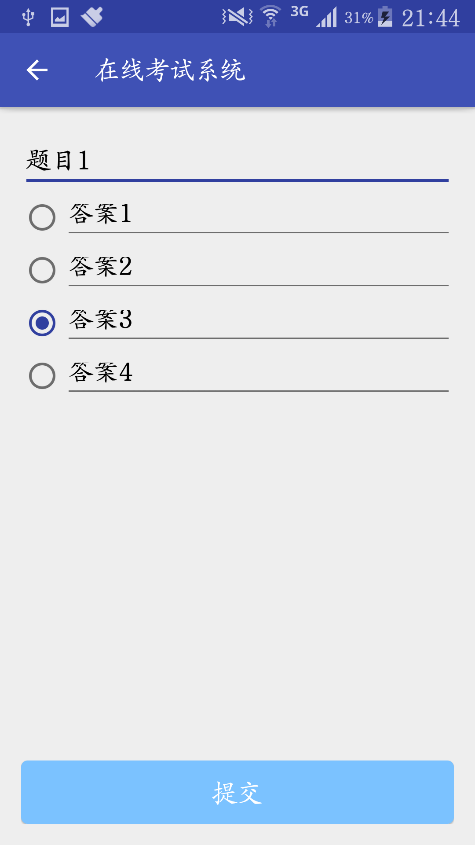
图5-26 试题编辑页面

图5-27 弹窗界面



5.3.4 查看学生成绩功能

教师通过主界面点击查看学生成绩，进入查看学生成绩功能，在这个界面，教师可以查看自己所管理的课程的所有学生的成绩。界面如图5-28：

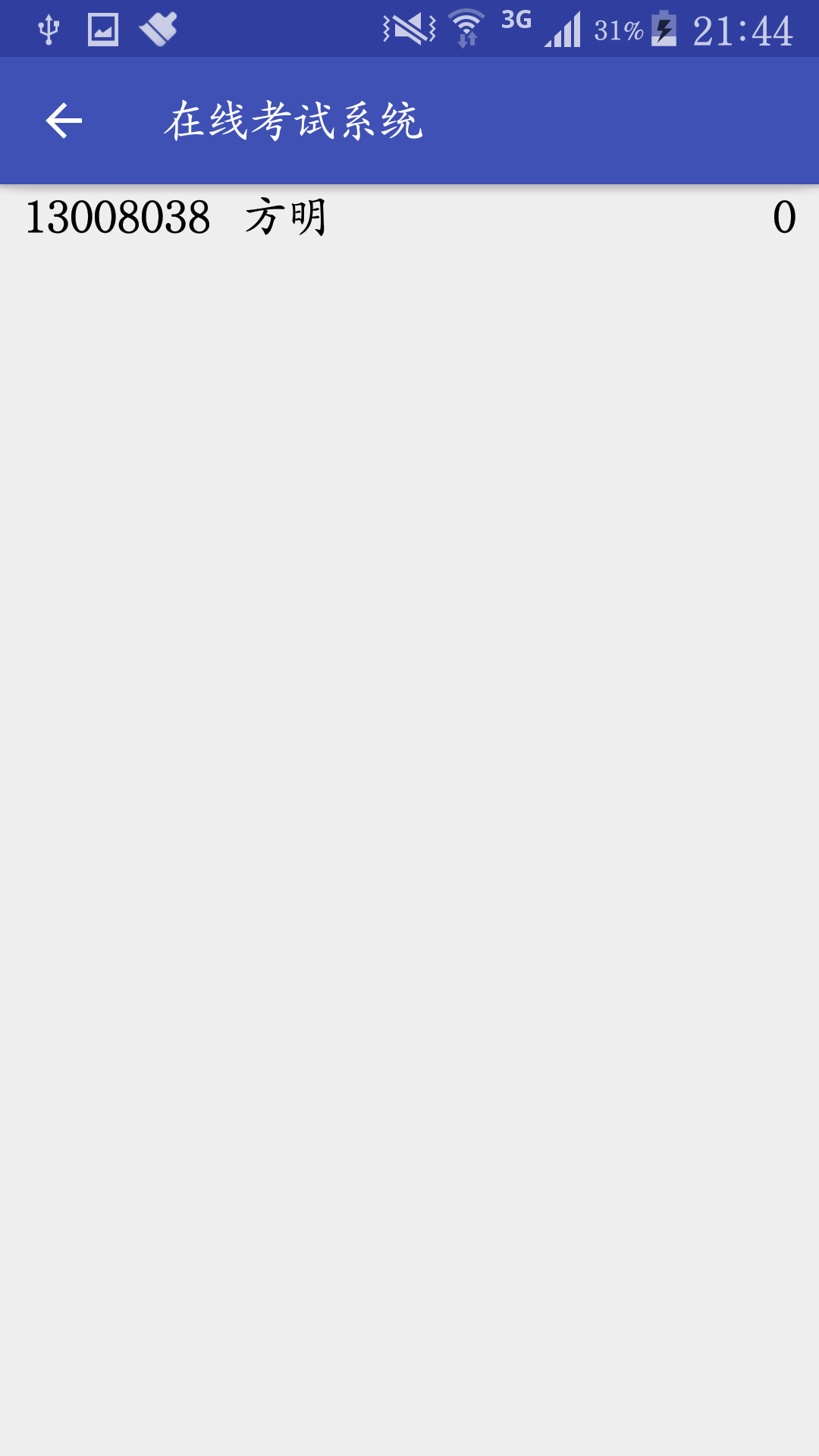


图5-28 查看学生成绩功能

服务端传输的JSON格式如下：

表5-9 获取成绩JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 获取成功 | {  "error": false,  "scorelist": [  {  "StudentID": "13008038",  "Score": "0",  "Name": "方明"  }  ]  } |
| 没有成绩 | {  "error": true,  "errormsg": "当前没有学生选该课程"  } |
| 获取失败 | {  "error": true,  "errormsg": "向服务器传输失败"  } |

5.3.5 修改信息功能

教师返回主界面点击修改信息按钮进入修改信息界面，修改信息界面显示当前教师的信息，为了保证教师信息的完整性，本系统仅允许教师修改电话和密码。教师点击提交后，系统会优先判断教师是否修改了成绩，避免不必要的提交，同时系统也会对输入的手机号码进行正则匹配，通过验证后向服务器提交。修改信息界面如图5-29：

图5-29 修改信息界面

服务端信息处理后返回的JSON格式如下：

表5-10 修改信息JSON格式

|  |  |
| --- | --- |
| 结果 | JSON结果 |
| 更改成功 | {  "error": false  } |
| 更改失败 | {  "error": true,  "errormsg": "向服务器更新失败"  } |
| 提交失败 | {  "error": true,  "errormsg": "向服务器传输数据失败"  } |

## 5.4 工具类

5.4.1 CheckboxListAdapter

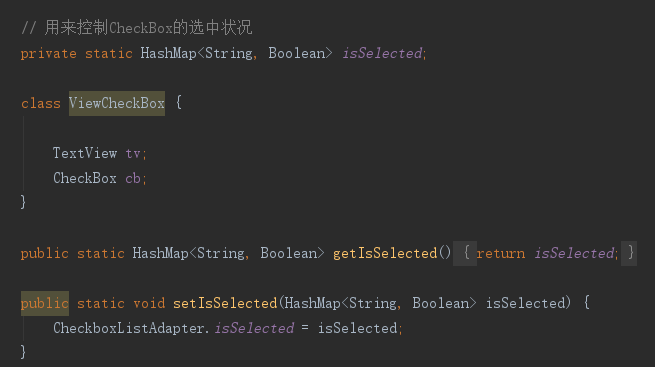
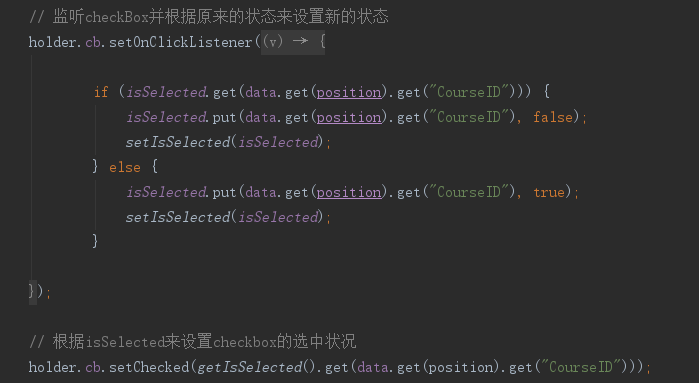
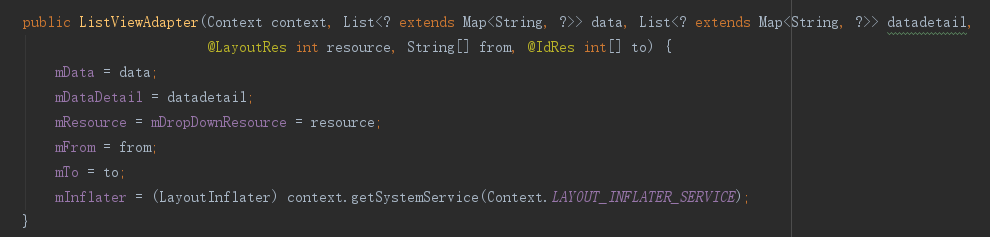
该工具类是为了实现学生选课的ListView的选择适配器。CheckboxListAdapter继承了BaseAdapter，设置一个静态属性isSelected记录当前指针对应的多选框的状态。主要代码如图5-30、图5-31：

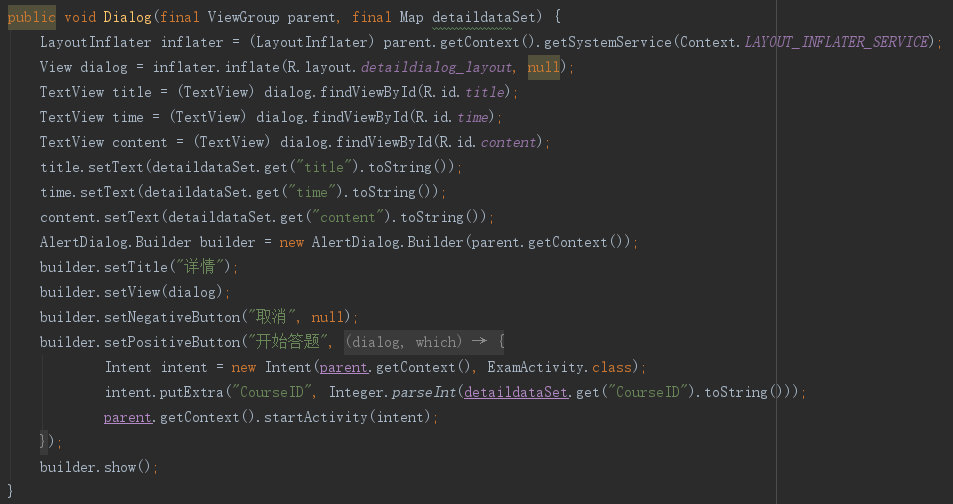
图5-30

图5-31

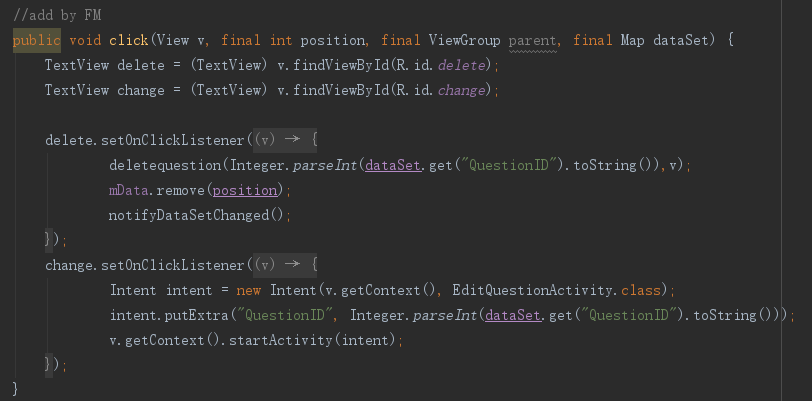
5.4.2 ListViewAdapter

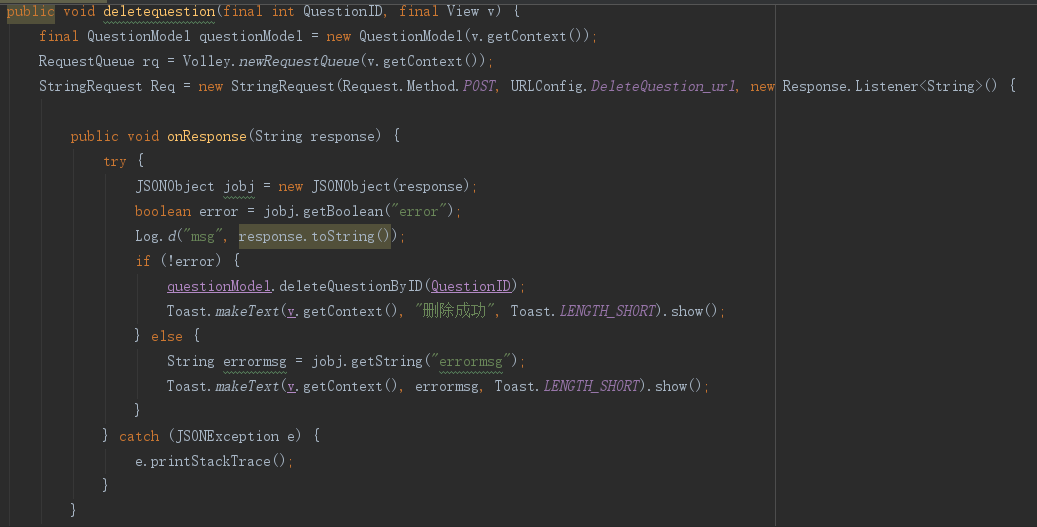
该工具类是为了实现学生答题界面ListView左滑菜单功能。该工具类的构造方法较于原来的BaseAdapter增加了一个List对象，为了实现左滑菜单的详情，记录试题的详情信息。主要代码如图5-32、图5-33：

图5-32

图5-33

5.4.3 TListViewAdapter

该工具类为了实现教师试题编辑页面ListView左滑菜单功能。该工具类增加了一个click方法，为左滑菜单的两个按钮作出不同的响应。工具类中也增加了一个删除方法，方法里记录了删除的试题编号，传输给服务器进行删除。主要代码如图5-34、图5-35：图5-34

图5-35

5.4.4 CircleImageView

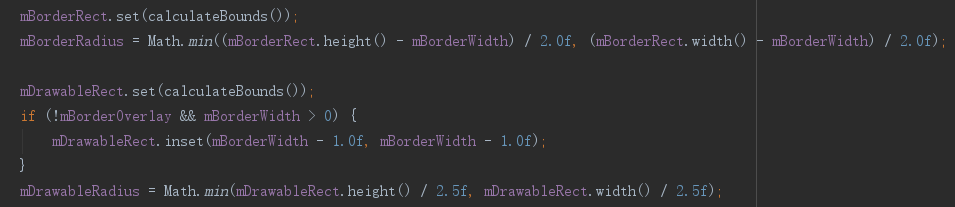
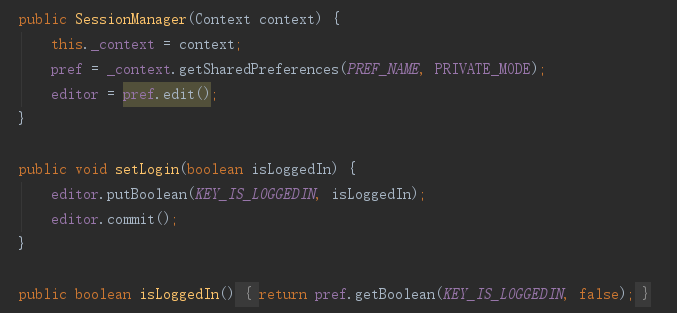
该工具类是为了实现用户头像圆形化。该工具类继承了ImageView，获取其中的图片地址，创建一个与背景色相同的图形，将图片置于中间，从而显现出圆形的效果。主要代码如图5-36：

图5-36

5.4.5 SessionManager

该工具类是为了记录用户的登录状态，控制用户进入本系统时，进入主界面或者登录界面。具体代码如图5-37：

图5-37

6 系统测试

## 6.1 系统测试的目的

系统测试的目的在于总结测试阶段对所有功能的测试、对测试结果进行分析，检查系统是否符合提出的需求。同时，在测试过程中，会发现软件中潜藏的错误以及漏洞，进行及时的更改，从而提高软件投入使用后的质量，减小软件风险。软件测试是用来评判一个软件的优劣性，为未来软件投入使用减少风险。因此软件测试对于开发一个新系统来说是必不可少的。

## 6.2 系统测试的过程

本系统采用的是边测试边开发的思想，利用Android Studio中的Android Tests(Junit)对必要的类进行测试。开发完成后对系统的各功能进行功能测试。

## 6.3 系统测试的功能点

学生模块功能测试：登录功能、注册功能、答题功能、查分功能、选课功能、修改信息功能。

教师模块功能测试：登录功能、试题查看功能、试题删除功能、试题增加功能、试题编辑功能、查看学生成绩功能、修改信息功能。

## 6.4 测试用例

6.4.1 学生模块的登录功能

表6-1 学生模块的登录功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 登录功能 |
| 输入 | {用户ID：13008888} |
| 预期结果 | 输入用户名和密码 |
| 实际结果 | 输入用户名和密码 |

表6-2 学生模块的登录功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 登录功能 |
| 输入 | {密码：123456} |
| 预期结果 | 输入用户名和密码 |
| 实际结果 | 输入用户名和密码 |

表6-3 学生模块的登录功能

|  |  |
| --- | --- |
| 序号 | 003 |
| 测试目的 | 登录功能 |
| 输入 | {用户ID：13008888，密码：123456} |
| 预期结果 | 不存在该账户 |
| 实际结果 | 不存在该账户 |

表6-4 学生模块的登录功能

|  |  |
| --- | --- |
| 序号 | 004 |
| 测试目的 | 登录功能 |
| 输入 | {用户ID：13008038，密码：456789} |
| 预期结果 | 密码错误 |
| 实际结果 | 密码错误 |

表6-5 学生模块的登录功能

|  |  |
| --- | --- |
| 序号 | 005 |
| 测试目的 | 登录功能 |
| 输入 | {用户ID：13008038，密码：123456} |
| 预期结果 | 登录成功，进入学生主界面 |
| 实际结果 | 登录成功，进入学生主界面 |

表6-6 学生模块的登录功能

|  |  |
| --- | --- |
| 序号 | 005 |
| 测试目的 | 跳转功能 |
| 输入 | 点击界面下方“没有账号？请注册” |
| 预期结果 | 跳转至注册界面 |
| 实际结果 | 跳转至注册界面 |

6.4.2 学生模块的注册功能

表6-7 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 注册功能 |
| 输入 | 空 |
| 预期结果 | 输入用户名和密码 |
| 实际结果 | 输入用户名和密码 |

表6-8 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 注册功能 |
| 输入 | {用户ID：13008888，密码：123456，确认密码：456789} |
| 预期结果 | 请确认密码 |
| 实际结果 | 请确认密码 |

表6-9 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 003 |
| 测试目的 | 注册功能 |
| 输入 | {用户ID：13008888，密码：123456，确认密码：123456} |
| 预期结果 | 请输入正确的手机号和姓名 |
| 实际结果 | 请输入正确的手机号和姓名 |

表6-10 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 004 |
| 测试目的 | 注册功能 |
| 输入 | {用户ID：13008888，密码：123456，确认密码：123456，姓名：abc，电话：18502410459} |
| 预期结果 | 请输入正确的手机号和姓名 |
| 实际结果 | 请输入正确的手机号和姓名 |

表6-11 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 005 |
| 测试目的 | 注册功能 |
| 输入 | {用户ID：13008888，密码：123456，确认密码：123456，姓名：方明，电话：12345678910} |
| 预期结果 | 请输入正确的手机号和姓名 |
| 实际结果 | 请输入正确的手机号和姓名 |

表6-12 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 006 |
| 测试目的 | 注册功能 |
| 输入 | {用户ID：13008038，密码：123456，确认密码：123456，姓名：方明，电话：18502410459} |
| 预期结果 | 已存在该账户 |
| 实际结果 | 已存在该账户 |

表6-13 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 007 |
| 测试目的 | 注册功能 |
| 输入 | {用户ID：13008888，密码：123456，确认密码：123456，姓名：方明，电话：18502410459} |
| 预期结果 | 注册成功 |
| 实际结果 | 注册成功 |

表6-14 学生模块的注册功能

|  |  |
| --- | --- |
| 序号 | 008 |
| 测试目的 | 跳转功能 |
| 输入 | 点击界面下方的“已有账号？请登录” |
| 预期结果 | 跳转至登录界面 |
| 实际结果 | 跳转至登录界面 |

6.4.3 学生模块的主界面功能

表6-15 学生模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 主界面功能 |
| 输入 | 空 |
| 预期结果 | 显示正确的当前用户信息 |
| 实际结果 | 显示正确的当前用户信息 |

表6-16 学生模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 主界面功能的侧滑功能 |
| 输入 | 点击“考试” |
| 预期结果 | 进入考试界面 |
| 实际结果 | 进入考试界面 |

表6-17 学生模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 003 |
| 测试目的 | 主界面功能的侧滑功能 |
| 输入 | 点击“查分” |
| 预期结果 | 进入查分界面 |
| 实际结果 | 进入考试界面 |

表6-18 学生模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 004 |
| 测试目的 | 主界面功能的侧滑功能 |
| 输入 | 点击“选课” |
| 预期结果 | 进入选课界面 |
| 实际结果 | 进入考试界面 |

表6-19 学生模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 005 |
| 测试目的 | 主界面功能的侧滑功能 |
| 输入 | 点击“修改信息” |
| 预期结果 | 进入修改信息界面 |
| 实际结果 | 进入修改信息界面 |

6.4.4 学生模块的考试功能

表6-20 学生模块的考试功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 考试功能 |
| 输入 | 没有进行选课的学生点击考试 |
| 预期结果 | 当前没有课程 |
| 实际结果 | 当前没有课程 |

表6-21 学生模块的考试功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 考试功能 |
| 输入 | 选完课程的学生点击考试 |
| 预期结果 | 显示已选课程的试题 |
| 实际结果 | 显示已选课程的试题 |

表6-22 学生模块的考试功能

|  |  |
| --- | --- |
| 序号 | 003 |
| 测试目的 | 考试功能 |
| 输入 | 选择一门课程左滑点击详情 |
| 预期结果 | 显示该课程的详情 |
| 实际结果 | 显示该课程的详情 |

表6-23 学生模块的考试功能

|  |  |
| --- | --- |
| 序号 | 004 |
| 测试目的 | 考试功能 |
| 输入 | 选择一门课程左滑点击答题 |
| 预期结果 | 跳转至该课程所有试题的答题界面 |
| 实际结果 | 跳转至该课程所有试题的答题界面 |

表6-24 学生模块的考试功能

|  |  |
| --- | --- |
| 序号 | 005 |
| 测试目的 | 考试功能 |
| 输入 | 滑至最后一道试题再左滑 |
| 预期结果 | 弹出确认提交框 |
| 实际结果 | 弹出确认提交框 |

表6-25 学生模块的考试功能

|  |  |
| --- | --- |
| 序号 | 006 |
| 测试目的 | 考试功能 |
| 输入 | 答完所有试题再左滑确认提交 |
| 预期结果 | 提交成功 |
| 实际结果 | 提交成功 |

6.4.5 学生模块的查分功能

表6-26 学生模块的查分功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 查分功能 |
| 输入 | 没有进行答题的学生点击查分 |
| 预期结果 | 当前没有成绩 |
| 实际结果 | 当前没有成绩 |

表6-27 学生模块的查分功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 查分功能 |
| 输入 | 进行答题的学生点击查分 |
| 预期结果 | 显示所有答题后课程的正确的分数 |
| 实际结果 | 显示所有答题后课程的正确的分数 |

6.4.6 学生模块的选课功能

表6-28 学生模块的选课功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 选课功能 |
| 输入 | 没有课程可选的学生点击选课 |
| 预期结果 | 当前没有课程 |
| 实际结果 | 当前没有课程 |

表6-29 学生模块的选课功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 选课功能 |
| 输入 | 学生点击选课选择课程 |
| 预期结果 | 选课成功，答题功能下有已选课程的试题 |
| 实际结果 | 选课成功，答题功能下有已选课程的试题 |

6.4.6 学生模块的修改信息功能

表6-30 学生模块的修改信息功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 修改信息功能 |
| 输入 | 不修改任何信息点击提交 |
| 预期结果 | 没有进行任何更改 |
| 实际结果 | 没有进行任何更改 |

表6-31 学生模块的修改信息功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 修改信息功能 |
| 输入 | 输入正确格式的电话和密码 |
| 预期结果 | 修改成功 |
| 实际结果 | 修改成功 |

6.4.7 教师模块的登录功能

表6-32 教师模块的登录功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 登录功能 |
| 输入 | {用户ID：13000000，密码：456789} |
| 预期结果 | 登录成功进入教师界面 |
| 实际结果 | 登录成功进入教师界面 |

6.4.8 教师模块的主界面功能

表6-33 教师模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 主界面功能 |
| 输入 | 无 |
| 预期结果 | 主界面显示正确的教师信息 |
| 实际结果 | 主界面显示正确的教师信息 |

表6-34 教师模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 主界面功能 |
| 输入 | 点击侧滑栏的题库按钮 |
| 预期结果 | 进入题库界面 |
| 实际结果 | 进入题库界面 |

表6-35 教师模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 003 |
| 测试目的 | 主界面功能 |
| 输入 | 点击侧滑栏的查看学生成绩按钮 |
| 预期结果 | 进入查看学生成绩界面 |
| 实际结果 | 进入查看学生成绩界面 |

表6-36 教师模块的主界面功能

|  |  |
| --- | --- |
| 序号 | 004 |
| 测试目的 | 主界面功能 |
| 输入 | 点击侧滑栏的修改信息按钮 |
| 预期结果 | 进入修改信息界面 |
| 实际结果 | 进入修改信息界面 |

6.4.9 教师模块的题库功能

表6-37 教师模块的题库功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 题库功能 |
| 输入 | 无 |
| 预期结果 | 显示当前教师所管理课程的所有试题 |
| 实际结果 | 显示当前教师所管理课程的所有试题 |

表6-38 教师模块的题库功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 题库功能 |
| 输入 | 点击一条试题的左滑菜单删除按钮 |
| 预期结果 | 该条试题被删除 |
| 实际结果 | 该条试题被删除 |

表6-39 教师模块的题库功能

|  |  |
| --- | --- |
| 序号 | 003 |
| 测试目的 | 题库功能 |
| 输入 | 点击界面右下角悬浮添加按钮 |
| 预期结果 | 增加一条新试题 |
| 实际结果 | 增加一条新试题 |

表6-40 教师模块的题库功能

|  |  |
| --- | --- |
| 序号 | 004 |
| 测试目的 | 题库功能 |
| 输入 | 点击一条试题左滑菜单的修改按钮 |
| 预期结果 | 进入修改该试题界面 |
| 实际结果 | 进入修改该试题界面 |

6.4.10 教师模块的查看学生成绩功能

表6-41 教师模块的查看学生成绩功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 查看学生成绩功能 |
| 输入 | 无 |
| 预期结果 | 显示当前课程已答题学生的成绩 |
| 实际结果 | 显示当前课程已答题学生的成绩 |

6.4.11 教师模块的修改信息功能

表6-42 教师模块的修改信息功能

|  |  |
| --- | --- |
| 序号 | 001 |
| 测试目的 | 修改信息功能 |
| 输入 | 不修改任何信息点击提交 |
| 预期结果 | 没有进行任何更改 |
| 实际结果 | 没有进行任何更改 |

表6-31 教师模块的修改信息功能

|  |  |
| --- | --- |
| 序号 | 002 |
| 测试目的 | 修改信息功能 |
| 输入 | 输入正确格式的电话和密码 |
| 预期结果 | 修改成功 |
| 实际结果 | 修改成功 |

7 结论

本系统主要实现了学生模块和教师模块。学生模块主要包括注册、登录、答题、查分、选课以及修改个人信息。教师模块主要包括对试题库的增删改查、对学生成绩的查看以及修改个人信息。在实现这些功能时，本系统利用了现如今比较主流的Android与PHP进行开发。开发中，也利用了Volley，这种比较新型的网络交互工具。在实现功能时，也自己编写了各种ListView的适配器，进行适配不同需求。基于Android的在线考试系统，较于现如今越来越多的基于浏览器的在线考试系统，Android端更为灵活，对于空间限制更小，是在线考试真正成为随时随地能够进行的事情。

# 后记

在设计这个基于Android的在线考试系统中，我学习了开发一个完整的Android系统的过程，从一开始的从无到有。从刚开始学习Android基本组件，到如今完成一个较为完整的在线考试系统，从中学习到了许多。在开发过程中，也查看了Android的ListView类、BaseAdapter类等类里面的方法，从而实现了一些新的功能。其中也不缺少曾经在校老师对我们的JAVA知识的巩固，从而才能到现在自我学习开发一些适合自己开发系统的东西。经过这个系统的开发，巩固了一个软件开发的流程，从需求分析到编码到测试，锻炼了自己，也学习了很多新知识。

# 参考文献

[1]王慧兰.基于Android平台的图书管理系统手机客户端开发[J].中外企业家，2016,11:204.

[2]陈红梅.基于Android的科目三模拟考试系统[J].智能计算机与应用，2016,02:55-57.

[3]李永宝，崔广章，陈琛，李岱英.基于Android的校园订餐系统[J].物联网技术，2016,04:71-75+78.

[4]江丽.基于android平台的实时互动远程教育系统设计与实现[J].信息通信，2016,04:121-122.

[5]黄晓先.基于Android的掌上校园交流系统设计与实现[J].开封教育学院学报，2016,03:280-281.

[6]王庆磊.Android移动数据安全探析[J].福建电脑，2016,03:101+109.

[7]邓昌友，肖遥，马小月，夏利，曾俊.基于Android智能手机数据安全的研究[J].福建电脑，2015,12:5-6.

[8]张中伟，苏静.基于云平台的Android移动学习系统设计[J].民营科技，2014,09:100+59.

[9] Bill Phillips, Chris Stewart, Kristin Marsicano. Android Programming: The Big Nerd Ranch Guide. (3rd Edition). Big Nerd Ranch, 2017.

[10] John Horton. Android Programming for Beginners. Packt Publishing , 2015.