

****

**本科毕业论文**

|  |
| --- |
| **基于Restful方式的在线学生信息管理系统** |

**全程**

**201231000826**

|  |  |
| --- | --- |
| 指导教师 | **熊敏 高级工程师**  **张义青 讲师** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学院名称 |  | **软件学院** | 专业名称 |  | **软件工程** |
| 论文提交日期 |  | **2016年4月15日** | 论文答辩日期 |  | **2016年4月24日** |

摘 要

随着社会的发展，用手机查询学生个人信息的需求不断地增强，针对这种情况，一些学校的教务网站开始将信息查询的功能转移到Android系列手机的客户端上。拥有了这种功能的Android软件使学生能够方便地查询到学生的个人信息。

本文基于REST技术，进行C/S结构的学生信息管理系统的开发。后台采用SpringMVC和Hibernate完成，通过和HttpClient技术完成数据传输；远程数据管理软件使用Intellij集成的Java Swing组件完成界面数据显示，能够远程管理托管在阿里云的服务器后台；安卓客户端使用Android Support Design Library中的组件完成界面设计，使用SQLite离线存储数据，通过访问远程数据库的内容查询学生个人信息。

本系统后台实现将数据库数据封装成Json数据，通过HTTP标准方法暴露出来的资源接口；数据管理软件设计并实现课程信息管理、公告管理、学生信息管理和学院信息管理四大模块，安卓客户端设计并实现学生个人信息和学院公告的查询模块。其中数据管理软件的课程信息管理可以管理每个学生和教师之间的课程关联，公告管理可以发布和管理各学院的公告，学生信息管理提供每个学生的账户信息管理，学院信息管理可以管理校方学院信息及其下属教师的个人信息；安卓客户端的学生个人信息查询可以查询所登入账户对应的详细课程信息及个人技能等级考试的相关信息，公告查询可以查看账户对应学院所发布的学院公告。

关键词：REST SpringMVC 信息管理

**RESTful Student Information Management System**

**Development and Application**

Quan Cheng

(College of Software, South China Agricultural University, Guangzhou, 510642, China)

**Abstract：**Nowdays, the damand of using a mobile phone to query personal information of students is enhanced.For this case,some school transferred their querying system onto Android phone so that do a faver to their students for school information query.

This paper is based on REST technology to finish the student information management sytem with C/S Architecture Model.This system is implemented by SpringMVC and Hibernate and transform its’ data by Json and HttpClient.The remote data management part use Java swing for GUI and its’ usage is manage the server’s data deployed in Alibaba’s mechine.Android client use the Android Support Design Library to finish view part and save local personal data from remote server by SQLite.

The system server part package MySQL’s data to json and send it to client by HTTP standard method. The remote data management part has these functions: course management,notice management, student management,academy management.The Android client has two module: persional information query and academy notice reminder.Course management can handler the course connection between students and teachers.Notice management provide interface for school to release notice and handler their notice history. Student information management provide interface for school to manage every student account’s infermation, and school can manage each students’ course data and so on.Academy information management can handler school’s academy list and all teachers belong each academy.Android client can let student login by their account and query the detail of personal course and technological exam.The client also provide the server to let student get the notice release from their academy.

**Key Words：** REST SpringMVC Information management

目 录

[1. 引言 1](#_Toc448701053)

[1.1. 选题背景 1](#_Toc448701054)

[1.2. 论文组织 1](#_Toc448701055)

[2. 相关技术与工具介绍 2](#_Toc448701056)

[2.1. REST 2](#_Toc448701057)

[2.2. SpringMVC 2](#_Toc448701058)

[2.3. HttpClient 3](#_Toc448701059)

[2.4. C/S模式 4](#_Toc448701060)

[3. 系统总体设计 5](#_Toc448701061)

[3.1. 系统概述 5](#_Toc448701062)

[3.2. 系统功能模块设计 5](#_Toc448701063)

[3.2.1. 客户端 5](#_Toc448701064)

[3.2.2. 服务器端 6](#_Toc448701065)

[3.3. 数据库设计 6](#_Toc448701066)

[3.4. 系统架构设计 9](#_Toc448701067)

[4. 系统实现 12](#_Toc448701068)

[4.1. 后台资源服务实现 12](#_Toc448701069)

[4.1.1. SpringMVC和Hibernate搭建Service框架 12](#_Toc448701070)

[4.1.2. HTTP标准方法和JSON实现数据传输 13](#_Toc448701071)

[4.2. 数据管理软件实现 14](#_Toc448701072)

[4.2.1. 可视化界面实现 14](#_Toc448701073)

[4.2.2. 远程数据库数据管理 15](#_Toc448701074)

[4.3. 安卓客户端实现 16](#_Toc448701075)

[4.3.1. 可视化界面实现 16](#_Toc448701076)

[4.3.2. 数据获取及保存实现 16](#_Toc448701077)

[5. 系统测试与运行 18](#_Toc448701078)

[5.1. 系统运行环境 18](#_Toc448701079)

[5.2. 系统运行与评估 18](#_Toc448701080)

[5.2.1. 数据管理软件 18](#_Toc448701081)

[5.2.2. 安卓客户端 23](#_Toc448701082)

[6. 总结与展望 27](#_Toc448701083)

[6.1. 总结 27](#_Toc448701084)

[6.2. 展望 27](#_Toc448701085)

[参考文献 28](#_Toc448701086)

[致谢 29](#_Toc448701087)

# 引言

## 选题背景

普通的学生信息管理系统通常需要学生使用个人电脑查询个人信息，这种做法一方面会消耗社会资源，另一方面会浪费学生的个人时间。而如果将一个信息管理系统的查询业务扩展到当前热门的智能手机中，用户群体就能够随时随地获取系统中的信息。

从学生角度分析，学生查询校方公告、考试成绩时往往需要使用个人电脑来完成，这会造成信息无法即时获取，也可能因为服务器端出现问题导致无法查询。从校方角度分析，只使用网页的方式向学生发布信息，学生会重复向服务器发送相同信息的查询请求，浪费网络资源，同时学生会在发布期末成绩等时段集体查询成绩，增大服务器的负担。

如今安卓系统技术已相当成熟，可以完美地完成信息交流过程中的运算和存储，缓解服务器端压力，同时其高普及率保证大部分学生能够及时获取相关信息。

本课题的研究意义在于通过以上提出智能手机接收信息、信息查询即时话等思路，设计并实现一个可行的在线学生信息管理系统，缓解学生不能即时获取校方信息的问题。

## 论文组织

本文将用六个章节阐述、设计并实现系统的过程、思想、技术，各章如下内容：

第一章：前言，介绍系统设计与实现的背景，展现文章的组织结构。

第二章：使用工具和关键技术介绍，对在系统设计和实现过程中使用到工具以及关键技术进行描述和分析。

第三章：系统设计与实现，对系统进行模块功能设计，数据库设计，系统架构设计，分析实现系统总体的技术要点与过程。

第四章：系统实现的功能的关键技术思路与实现，关键代码分析。

第五章：系统最终运行效果展示，图文展示验收最终系统实现运行起来的效果。

第六章：总结与展望。

# 相关技术与工具介绍

## REST

REST是一种能够使用URI将一系列资源链接起来的设计风格（Pongsathorn Noimanee，Yachai Limpiyakorn，2009）。REST指定了一组架构约束条件和原则，满足这些要求的系统可以称为RESTful。REST使用URI实现资源定位，从这个角度上讲，对外提供REST式的Web服务就是公布一系列的URI及其参数，这使得REST的实践过程简单到了极致（韩陆，2015）。

REST主要原则包括：对资源使用一致的命名规则，即使用URI标识所有需要标识的事物，特别是应用中需要使用到的资源；链接所有资源事物，系统中的资源都可以使用特定的URI获取；使用HTTP标准方法，使客户端程序能与系统资源相互协作，使用POST、DELETE、PUT、GET完成数据管理中的增删改查；资源多重表述，为同一资源提供多种表述形式，提高系统重用性和可扩展性；无状态通信，通信状态应该由客户端负责维护，服务端应该返回无状态的资源信息，客户端和服务器端只能通过相互发送遵循预定义协议的消息来进行交互（Leonard Ricbardson，2014）。

RESTful架构约束主要有使用CS模式，通信只由客户端发起，服务端表现为根据请求进行响应的模式；内容缓存，响应内容或者资源可以在通信链进行缓存，可以改善网络效率，在本系统的表现为数据可以在安卓手机离线保存，同时利用安卓手机减轻服务器运算负担；分层系统，通过规定组件的行为，使其只能接受与其交互的紧邻层的信息，达到将服务分解为若干层的效果；以及统一接口、无状态和按需编码。

## SpringMVC

Spring MVC属于SpringFrameWork的后续产品，提供了构建 Web 应用程序的全功能 MVC 模块。框架主要由DispatcherServlet、HandlerMapping、Controller、ViewResoler、视图组成，将对数据库的操作分为了界面视图部分，业务逻辑模型部分，数据库通信控制部分，对层次的分隔强、耦合低（许文稼，赵英凯，2008）。



图1 REST技术核心组件

组件细节如图1所示：DispatcherServlet是SpringMVC的核心，其本质是一个Servlet，通过配置文件中的<servlet-mapping>完成对request的监听，用以控制程序的业务流程（Praveen Gupta，2010）。当DispatcherServlet接收到请求时，会根据HandlerMapping的内容，调用Controller对请求进行处理（舒礼莲，2013）。在本系统的表现形式为，系统接收到资源请求，会调用对应的Controller返回相关资源。

## HttpClient

HttpClient 是一个可以用来提供高效的、最新的、功能丰富的支持 HTTP 协议的客户端编程工具包，并且支持 HTTP 协议最新的版本和建议（王超，闾陈莉，2014）。HttpClient实现了所有HTTP的方法，可以方便地配合REST架构约束中的使用HTTP标准方法的要求，同时支持HTTPS协议。使用HttpClient需要以下步骤：首先创建HttpClient实例，接着创建某种连接方法的实例，这里包括HttpGet、HttpPut、HttpPost、HttpDelete等，HttpClient中的execute方法可以调用以上实例，完成HTTP中的方法。接着读取HTTP方法的response，释放连接，最后对得到的内容进行处理。

HttpClient同时支持多种运用场景，同时能够加快系统效率以及开发效率，本系统中的数据管理软件以及安卓客户端均使用HttpClient完成HTTP请求。

## C/S模式

Client-Server是为了配合REST技术以及安卓开发所选用的系统结构。通过它可以充分利用两端硬件环境的优势，将任务合理分配到Client端和Server端来实现，降低了系统的通讯开销。根据功能分布的原则，客户端完成数据处理，数据表示以及用户接口功能；服务器端完成数据库管理系统的核心功能。

C/S结构优点为充分发挥客户端处理能力，将部分业务分配到客户端进行处理，客户端可以存储部分数据，保证系统响应速度，同时减轻网络负担。C/S模式是一种可扩展的、将系统分布在不同物理设备上的系统结构（C. Bruce Kavan，2016），使用这种系统结构能够降低增加新Server或者服务器主机的成本。但Client/Server结构的系统需要很大的运维和更新成本（Julia Janicki，2016）。具体缺点在于需要安装客户端、无法跨平台和高成本。用户使用系统时都需要安装客户端，若系统更新或出现问题都需要对客户端进行更新；同时不同操作系统需要不同的客户端，大大增加系统开发的工作。此外，需要考虑客户端和服务端数据同步问题，这可能意味着消耗大量的网络资源。

# 系统总体设计

## 系统概述

本系统设计成一个可以让学生查询校方信息的平台，可以根据自身学号获取本人课程的相关信息，当前类似四六级等级考试的专业技能等级考试信息以及校方发布的公告。而校方可以远程管理托管在阿里云的学生数据，进行管理学生账号、发布公告和学生成绩等操作。

根据REST技术的特点，本系统采取版本迭代的方式，不断推出新版本，在系统框架确定的前提下不断为系统添加新功能，让系统功能逐步完善。

## 系统功能模块设计

通过本系统，在移动端实现学生的登入登出、个人信息以及学院公告的查询，在SERVICE端实现学院信息、学生个人信息以及课程信息的管理，并通过数据库保证数据同步。图2为本系统功能模块图。



图2 系统功能模块图

### 客户端

客户端功能如下：

登入：学生根据系统中预设的账户进行登入操作，可以根据账户查看不同学生的信息。

登出：已登入账户可以进行登出操作，同时删除相关的学生信息。

更新相关课程信息：根据登入的学生账户，可以更新并离线保存该学生当前的课程信息、成绩等。

更新学院发布的公告：已登入的学生可以获取学院发布的公告，相同学院的学生会收到同一份公告。

更新技能等级考试信息：根据登入的学生账户，可以更新并离线保存该学生当前的技能等级考试信息。

数据显示：学生在已登入的情况下可以直接查看离线保存的数据。

### 服务器端

学生账户管理：可以管理学生账户信息，包括密码、学院分组等。

课程管理：可以进行建立课程，更新考试成绩等一系列操作。

学院数据管理：可以对学校当前学院及其下属教师的信息进行数据管理。

技能等级考试管理：可以根据已有的学生账户，管理个人技能等级考试信息。

## 数据库设计

本系统使用MySQL5.6以及SQLite数据库，设计的数据库有6张表，其中包括notice表、academy表、teacher表、student表、course表、technologicalexam表。

其中，表与表之间关系如下所述：

academy与notice是一对多关系。

academy与teacher是一对多关系。

academy与student是一对多关系。

student与technologicalexam是一对多关系。

student与course是一对多关系。

teacher与course是一对多关系。

主要的表设计与关系如图3所示：



图3系统数据库设计

在本系统中，涉及到的数据项和表结构如下：

表1 学院信息表——academy

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段意义 | 类型 | 长度(字符) |
| academyId | 学院编号 | varchar | 255 |
| academyName | 学院名称 | varchar | 255 |
| academyAddress | 学院地址 | varchar | 255 |

表2 学生信息表——student

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段意义 | 类型 | 长度(字符) |
| studentId | 学生账户 | varchar | 255 |
| studentPassword | 登入密码 | varchar | 255 |
| sex | 性别 | varchar | 255 |
| identityCard | 身份证 | varchar | 255 |
| bankCard | 银行卡号 | varchar | 255 |
| academyId | 所属学院id | varchar | 255 |

表3 课程信息表——course

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段意义 | 类型 | 长度(字符) |
| courseId | 课程编号 | varchar | 255 |
| credit | 学分 | double | 16 |
| teacherId | 教师id | varchar | 255 |
| studentId | 学生id | varchar | 255 |
| courseName | 课程名 | varchar | 255 |
| courseTime | 上课时间 | varchar | 255 |
| courseDate | 学期 | varchar | 255 |
| test1 | 平时测试1 | double | 16 |
| test2 | 平时测试2 | double | 16 |
| test3 | 平时测试3 | double | 16 |
| exercises1 | 练习题1 | double | 16 |
| exercises2 | 练习题2 | double | 16 |
| exercises3 | 练习题3 | double | 16 |
| exercises4 | 练习题4 | double | 16 |
| exercises5 | 练习题5 | double | 16 |
| finalTest | 期末考试成绩 | double | 16 |
| dailyMark | 平时分 | double | 16 |
| sum | 总分 | double | 16 |

表4 学院公告表——notice

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段意义 | 类型 | 长度(字符) |
| noticeId | 公告编号 | varchar | 255 |
| noticeTitle | 公告标题 | varchar | 255 |
| noticeContext | 公告内容 | varchar | 255 |
| noticeOperator | 发布者 | varchar | 255 |
| academyId | 所属学院id | varchar | 255 |

表5 教师信息表——teacher

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段意义 | 类型 | 长度(字符) |
| teacherId | 教师编号 | varchar | 255 |
| teacherName | 教师名称 | varchar | 255 |
| teacherDepartment | 所在部门 | varchar | 255 |
| teacherStatus | 具体职业 | varchar | 255 |
| academyId | 所属学院id | varchar | 255 |

表6 技能等级考试表——technologicalexam

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段意义 | 类型 | 长度(字符) |
| tId | 编号 | varchar | 255 |
| tName | 考试名称 | varchar | 255 |
| tDate | 考试日期 | varchar | 255 |
| tSorce | 考试分数 | double | 16 |
| studentId | 所属学生id | varchar | 255 |

## 系统架构设计

为配合REST核心技术以及Android系统特性，本系统采用C/S结构，资源接口编程设计、MVC架构，提高了系统的重用、可扩展性、并发性、运行速率。图4为本系统的架构设计:



图4系统架构设计

后台采用MVC思想，用Hibernate技术奖表映射出Entities，将常用的数据库操作代码设计为Data Access Object，用SpringMVC将具体业务逻辑运算的代码封装在Service层，Resource Interface层暴露出REST的API，使其他程序能使用HTTP标准方法读取和管理MySQL中的数据。前端分别使用Android开发技术和Intellij IDEA集成的JAVA swing开发技术实现一个适用对象为学生的Client端和个人电脑端的ServiceGUI端，能够遵循RESTful标准管理远程后台的相关信息。

根据RESTful标准，后台搭建HTTP标准方法的资源接口，即使用HttpClient技术实现GET方法获取数据，DELETE方法删除数据，POST方法新增数据，PUT方法更新数据的HTTP标准方法。数据传输方面，本系统使用Alibaba的技术完成数据的封装，用Google的Gson技术完成解读，同时使用URLEncoder和URLDecoder实现跨平台数据传输。这些技术保证了系统的重用性，可扩展性和足够高的运行效率。

安卓Client端使用Google官方Android Design Support Library中的Tablayout和Viewpage完成可视化组件，构建出类似新浪微博手机版的界面，可以划屏切换学院公告、课程信息和技能等级考试的不同信息，同时使用HttpClient发送请求和接受Json数据。接收到数据后，使用SQLite保存数据，使学生能够离线查看自身信息。数据以远程数据库为标准，每次更新信息都会获取远程数据库的相关数据保存到本地。

服务器GUI的界面部分使用Intellij IDEA集成的Java Swing开发工具，使用xml文件对Java可视化组件进行组建和一些Windows平面风格的改进，提高了Java可视化组件的用户体验。数据传输部分使用即时更新的方式管理远程数据库数据，不保留本地数据。即每次修改保存都会直接更新到数据库，以数据库为基准值保持数据同步。

在实际应用场景中，ServiceGUI不会直接影响Android Client，校方只能通过ServiceGUI更新信息到远程数据库，学生使用安卓手机获取远程数据库当前的信息。

# 系统实现

## 后台资源服务实现

### SpringMVC和Hibernate搭建Service框架



图5系统后台框架

如图5所示，后台使用SpringMVC和hibernate搭建从MySQL数据库到Resource Interface的框架。首先使用hibernate生成集成了JDBC的DAO和与MySQL相匹配的Entities，这使得系统能从代码角度修改MySQL的数据。其次，使用SpringMVC管理系统内的组件，即使用Spring注解实现以springframework中的工厂为管理器，DAO、Entities、Service和Resource Interface为工厂中的bean为被管理对象，自动实现beans的生成与使用。Service和Resource Interface的实现使用了Spring工厂生成的beans，其中Service处理具体的业务逻辑计算，Resource Interface用于暴露RESTful的数据处理接口。当系统监听到外部通过Resource Interface发送请求时，会调用Service进行相关操作，Service则调用DAO进行具体操作或者获取相关数据并通过Resource Interface返回到外部。

本节实现难点有以下几点：Hibernate自动生成的文件不符合系统要求、SpringMVC尚未支持RESTful的Resource注解和Spring工厂的配置问题。

Hibernate自动生成的实体类中，每个实体都会根据数据库外键生成一系列的set，若直接调用并生成json数据会出现循环，及一个实体中的set会包含其外键关联的实体，而其他实体又会包含对应的set。这种循环会对系统造成很大的影响，因此系统去除了实体类中的set。同时这些正确的实体类重用到了另外两部分软件中。

缺乏支持RESTful约束的软件开发框架使得开发一个完整RESTful Web服务成为一种挑战（Mou’ath Hourani，Qusai Shambour，2014），SpringMVC为使用者提供了Controller注解，用于注解服务器所暴露的接口，但RESTful的服务应该暴露Resource型的接口，因此这里还是使用了Controller注解来完成资源接口的功能。

Spring工厂applicationContext.xml来完成工厂配置，但SpringMVC中还增加了一个spring-mvc.xml来配置，这种情况下，可能会出现一些配置的重复，或者改写了正确配置，很容易出现系统错误。另外，配合Hibernate的SpringMVC工厂需要配置transactionManager的bean才能确保系统能正常运行。

### HTTP标准方法和JSON实现数据传输



图6 系统数据传输方法

根据REST技术，Service端对资源使用一致的命名规则，外部通过HTTP标准方法调用URL，即可获得对应资源。系统Service端挂载在阿里云服务器中，资源服务器URL为http://115.28.167.136:8080/RESTSIQS/\*，个人电脑及安卓手机会根据HTTP标准方法向服务器发送请求，若使用GET方法则会返回服务器当前数据，若使用DELETE、POST和PUT方法则会返回数据更新后的新数据。使用这种方式实现数据传输能够严格遵循REST核心技术的规范，这会使得系统重用性大大提高。若系统功能发生变动，系统可以在不改动原有服务的情况下release。

系统实现难点在于：JSON传输中出现的乱码问题。出现乱码是一个复杂度很高的BUG。首先，系统需要确保数据库、Android端以及PC端软件使用的字符集是统一的，用户在更新数据时输入的中文很容易以乱码形式存储在数据库中，这样会毁掉数据库中的数据。其次，系统使用了URLAccepter类中的方法编译和反编译传输的数据，否则数据中的特殊符号会传输失败。

## 数据管理软件实现

### 可视化界面实现

数据管理软件是原生JAVA组件构建的JAR可运行程序。不同的是可视化界面部分由Intellij IDEA集成的工具完成。系统每个窗口都是Intellij IDEA定义的GUI form，每个GUI form由一个java文件和一个form文件组成，并由form文件决定组件布局，java文件完成业务逻辑计算。

具体实现过程为：使用MainView.form完成数据管理软件中AcademyPanel、CoursePanel、NoticePanel和StudentPanel的布局，在自定义四个Panel的具体业务逻辑，搭配相关数据的管理Dialogs完成整个可视化界面。AcademyPanel中以双重列表完成数据显示，左侧为学校当前的学院列表，右侧为被选中学院中的教师列表，通过双击等操作即可管理远程数据库中的数据。CoursePanel为关联教师与学生之间课程信息的界面，会以多对多的形式显示两者之间的关联。StudentPanel显示当前学生账号信息，双击一个学生账号时可以管理该账号对应的基本信息、技能等级考试信息和该账号对应的课程信息。NoticePanel为学院查看和管理当前公告的界面，不同的学院发布的公告会被对应学院的学生接收到。若系统功能发生变动，软件可以在主界面添加新的tab界面，同时由于各功能已经被tab分割，因此进行开发重构时只需要在原有基础上新增或更新功能，不会对系统框架造成影响。

这里有一个实现难点是在table中添加edit和delete按钮。为了实现这个功能，软件使用了ButtonRenderer类以及table listener。对于需要添加button的table，软件在实现过程中对table的列进行判断，为edit和delete列添加ButtonRenderer，使其在界面上显示为按钮形状；其次，使用listener判断列的单击事件，若用户点击了edit或者delete列，界面就会跳转至对应的界面。

### 远程数据库数据管理

实际应用场景中，用户在程序界面完成数据更新后，会即时保存在远程数据库中并更新本地程序的数据。系统数据管理软件会向Service发送与Resource Interface相对应的请求，并获取新的数据，即在使用GET以外的方法后，程序都会调用一次GET方法获取新的数据。

以更新数据为例，用户更新完某段数据后，程序首先将新数据封装成一个pojo对象，其后转化为JSON数据，通过HttpClient中的HttpPut将请求发送到数据库，而后程序会调用一次HttpGET方法获得新的数据，并显示在可视化界面中。

public static void put(String URL, Object object) throws ConnectException {  
 try {  
 String jsonObject = JSONObject.*toJSONString*(object);  
 jsonObject = URLAccepter.*encrypt*(jsonObject);HttpClient httpclient = new DefaultHttpClient();  
 HttpPut request = new HttpPut(URL + jsonObject);  
 httpclient.execute(request);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }

功能实现难点在于数据的实时更新以及数据合法性问题。由于软件不保存本地数据，同时要实现立即查看所更新数据的需求，软件会在对数据库进行POST，DELETE，PUT操作后进行一次GET操作。在这里的一个实现方法使用到了REST的一个优势：资源分离。每种资源都是分开保存的，因此每次软件更新了数据库后，软件会只获取对应的新数据而不是刷新整个系统的数据，这可以减轻Client和Service端的负担。数据合法性指的是用户在进行界面操作时，会进行与软件功能不一致的操作，譬如打开可以更新同一资源的两个窗口，或者对一些必要的数据进行留空操作，当然这里也包括了中文输入的需求实现。为了处理这些情况，系统首先加入了界面锁的设定，以树状操作的思想引导用户不要对单一资源开启多个更新窗口。其次，系统对有数据输入的界面进行了非空判断，若出现了空输入，会发出提示并且停止更新操作，直到用户按规定输入数据或者选择退出。而中文输入问题牵涉到前文提到的前后端字符集统一问题，因此系统重构了一部分数据库以及界面数据相关的代码，使用GBK编码完成数据的保存和更新。

## 安卓客户端实现

### 可视化界面实现

安卓端主要使用android design support library实现界面，在主界面使用类似微博的滑动切换效果实现数据显示。单击显示列表中的内容会显示对应的详细信息。

当用户新安装或者未登入时，打开程序会首先进入登入界面，系统对其输入的账号密码进行判断，若登入成功，会获取账户对应的信息并呈现在主界面上，主要信息有学院公告、学生课程信息和该学生所进行过的技能等级考试。单击学院公告会显示改公告的详细信息，包括标题和公告内容等；单击课程信息会显示该课程的上课时间、练习成绩、平时测试成绩、总成绩等内容；技能等级考试会直接在列表显示所有信息。若用户已登入，再次进入程序时，系统会调用手机离线保存的数据，可以不通过网络直接显示手机存储的信息。

用户可以在程序菜单栏选择注销用户、更新信息和直接退出程序。注销用户时，界面会返回登入界面，同时删除所有已保存的账户信息；更新信息是在系统登入的情况下，直接根据账户获取远程数据库中的数据，达到安卓客户端同步数据的效果。

系统同时为系统更新留下了大量空间，开发者可以遵循原有的android design support library添加新的组件，包括瑞士军刀式界面、悬浮按钮或者增加Tablayout中的组件。

### 数据获取及保存实现

系统同样使用HttpClient完成数据请求。

public static JSONObject get(String URL) {

JSONObject jsonObject = null;

try {

HttpEntity entity;

entity = null;

HttpClient httpclient = new DefaultHttpClient();

HttpGet request = new HttpGet(URL);

HttpResponse response = httpclient.execute(request);

entity = response.getEntity();

String result = EntityUtils.toString(entity);

jsonObject = (JSONObject) JSON.parse(result);

}catch (IOException e)

{

Log.i("devouty\_HTTPJSONGetter",e.getMessage());

}

return jsonObject;

能够实现这个系统重用的原因是HttpClient和alibaba.均支持多系统多平台。与数据管理软件不同的是，安卓系统的HttpClient只使用了HTTP中的GET方法，这意味着安卓端无法改动数据库的数据。

数据的离线存储使用了安卓系统自带的SQLite数据库，通过deploy远程数据库的数据到本地，可以解决智能手机在某些情景下网络不佳的问题，同时由于用户查询系统数据所消耗的时间只包含打开程序所消耗的时间，所以使用本地数据库能够大大地提高用户体验。具体步骤为：当接收到更新请求时，系统首先根据账户组织URL向远程数据库请求数据，接收到json格式的数据后，清空旧版本的数据，然后将json数据转化为SQLite能够处理的ContentValues，实例化SQLiteDatabase并将数据保存到SQLite中，最后关闭数据库。

关键代码：

public void saveNotice(JSONObject jsonObject) {

JSONArray jsonArray = jsonObject.getJSONArray("result");

int length = jsonArray.size();

JSONObject obj;

ContentValues cv = new ContentValues();

SQLiteDatabase database = DatabaseUtil.this.getWritableDatabase();

database.execSQL("delete from notice;");

for (int i = 0; i < length; i++) {

obj = (JSONObject) jsonArray.get(i);

cv.put("noticeId",obj.getString("noticeId"));

cv.put("noticeTitle",obj.getString("noticeTitle"));

cv.put("noticeContext",obj.getString("noticeContext"));

cv.put("noticeOperator",obj.getString("noticeOperator"));

cv.put("academyId",obj.getString("academyId"));

Object t = database.insert("notice", null, cv);

}

}

# 系统测试与运行

## 系统运行环境

使用Xshell5登录公网IP为115.28.167.136的阿里云服务器，安装MYSQL5.6并建立对应的数据库，并将程序包RESTSIQS.war部署到Apache tomcat目录的{ApacheTomcatHome}/webapps中，启动MySQL和tomcat服务。

指令：

cp RESTSIQS.war {ApacheTomcatHome}/webapps

{ApacheTomcatHome}/bin/start.sh

/bin/systemctl start mysqld.service

系统思路为使用阿里云服务器远程运行Service，将数据和服务挂载在远程Linux机中，用户分别使用安卓手机和PC端数据管理软件读取和更新远程Linux机的数据。这么做的优势是利用企业服务确保系统能够长时间地服务，同时不用担忧学校物理环境对系统造成的影响。

## 系统运行与评估

### 数据管理软件

1. 课程信息管理

如图7所示：

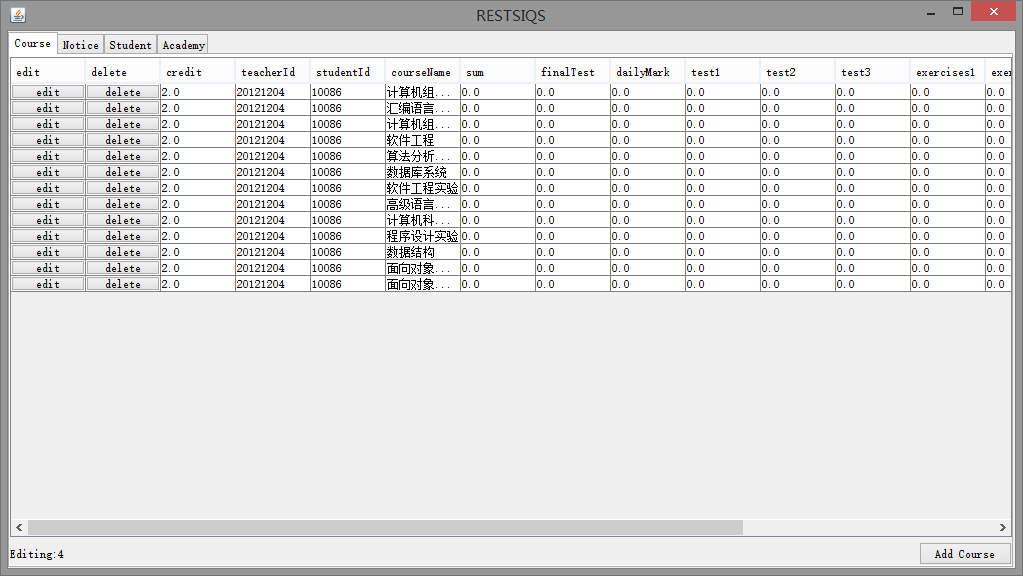


图7 课程列表信息界面

课程管理界面可以直接管理教师与学生之间的课程联系，并记录每个学生的单个课程成绩。打开程序，数据管理软件会直接读取远程数据库当前的数据并以列表显示在界面上。由于单个课程的内容过多，因此该界面不支持直接在table组件修改课程数据，列表界面只支持滑动显示或者选择更新操作，对于具体的数据更新操作，用户应该点击edit键或者双击其中一行来进行查看和更新操作。拖动表头可以变动数据显示区域，拖动滑动栏可以显示隐藏的数据，点击edit按钮或者双击一个课程，可以显示该课程的详细信息：

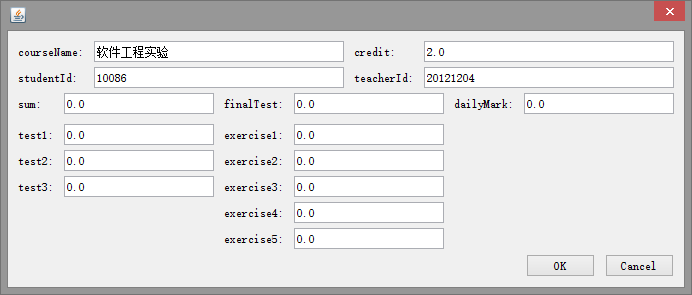


图8 课程信息界面

输入课程成绩点击OK键即可向远程数据库发送更新请求，课程信息界面关闭的同时会刷新显示远程数据库当前的数据。数据输入栏不应该为空，否则会在远程数据库保存默认值，其中courseName、studentId、teacherId默认值为空，credit及分数项默认值为0。

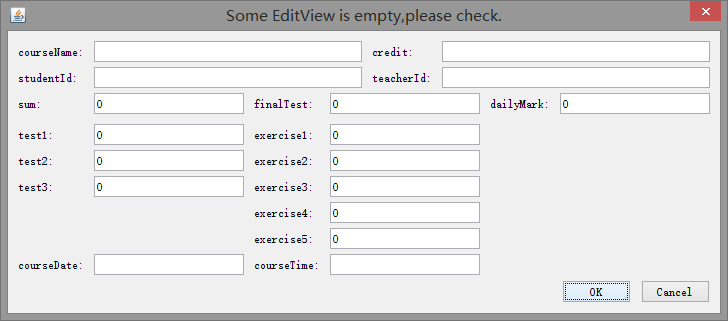


图9 新增课程界面

点击新增课程，显示新增课程界面，用户输入课程名、学分、对应的学生和教师账号即可新增课程，由于这几项是必填项，数据项缺失会导致课程信息不能正确关联，相当于数据丢失，因此设置了非空判断。这里与编辑课程界面的不同点在于多出courseDate和courseTime，分别代表该课程的学期和该课程每周的上课时间。

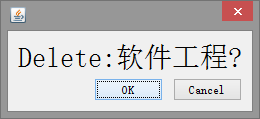


图10 课程删除界面

此外，点击列表中的delete按钮可以进行删除课程的操作，确认删除后会同步更新到远程数据库。若点击取消、点击右上角的关闭按钮或者按Esc键，会取消删除，返回原课程列表页面。

1. 公告管理

公告管理界面可以向校方提供一个发布和管理学院公告的操作界面，它可以分学院向对应的学生发布公告。不同学院的公告根据每个公告对应的学院id来划分，可以通过点击edit键或者双击进行编辑，通过点击delete键进行删除操作。

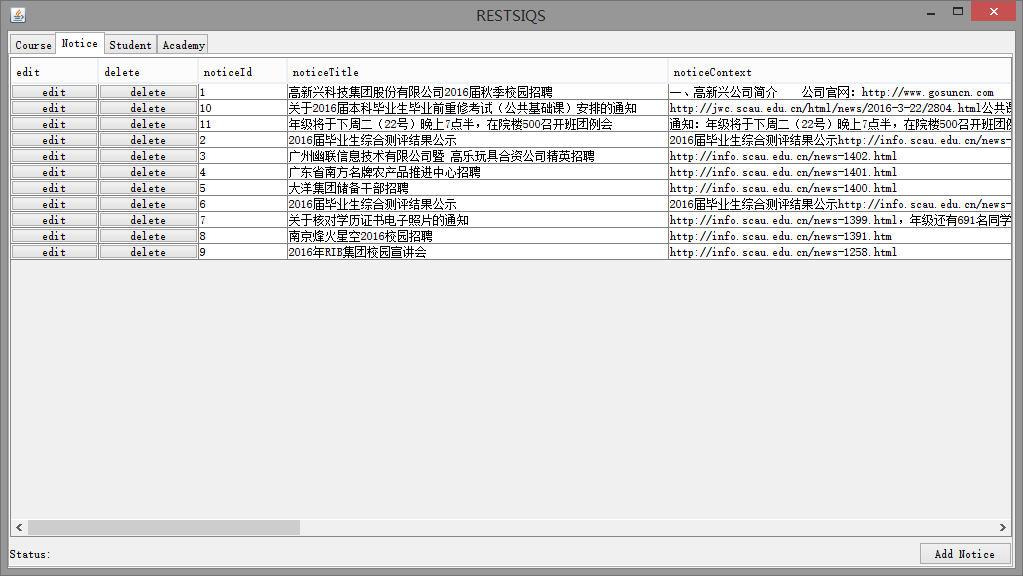


图11 公告列表界面

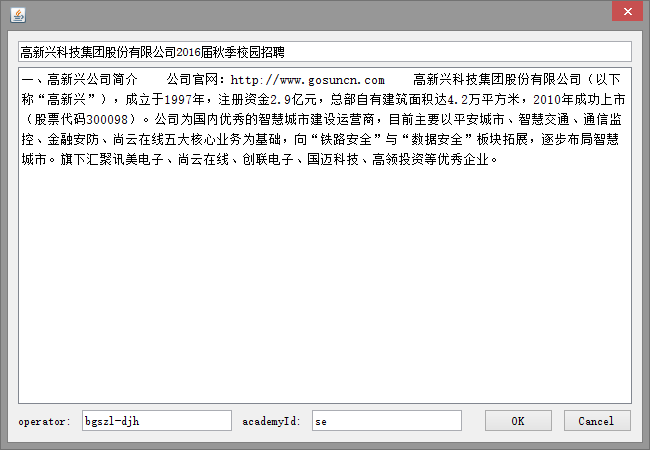


图12 公告编辑界面

点击新增公告界面，可以发布新公告，输入对应的公告信息及学院id并确认，该学院下的学生可以在手机端接受到本学院的公告。界面支持第三方输入法进行输入，其中最上方输入栏为公告标题，中央输入框为公告详细内容， operator为发布公告的操作者署名，academyId的功能是根据id划分公告对应的学院，这将决定了哪个学院的学生可以查看到改公告，点击OK键会保存并发布到学生手机中。

1. 学生信息管理

如图13所示：

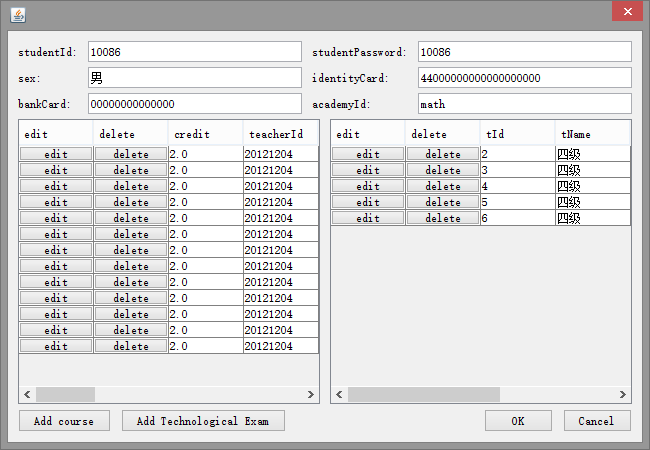


图13 学生个人账户管理界面

学生信息管理界面主要管理每个学生账户的信息，其中包含个人基本信息、账号、密码以及该学生的课程、技能等级考试信息。两个列表都可以双击编辑该学生的学习信息，其中课程信息与课程管理是统一数据的。其中studentId为账户名，账户名不得重复否则无法保存，studentPassword为学生登入手机端的密码。下方两个表格操作方式与其他功能类似，差别是当管理者对这两个表进行更新操作并放回到个人账户管理界面时也会进行学生数据的联网保存。

在后台数据方面，这个界面显示了三种资源的内容。以学生账户为主键，会分别显示student、course和TE的相关内容。当然，用户进行一些界面操作并选择保存数据时，远程数据库能够修改多个资源的内容。

1. 学院信息管理

界面如图14所示：

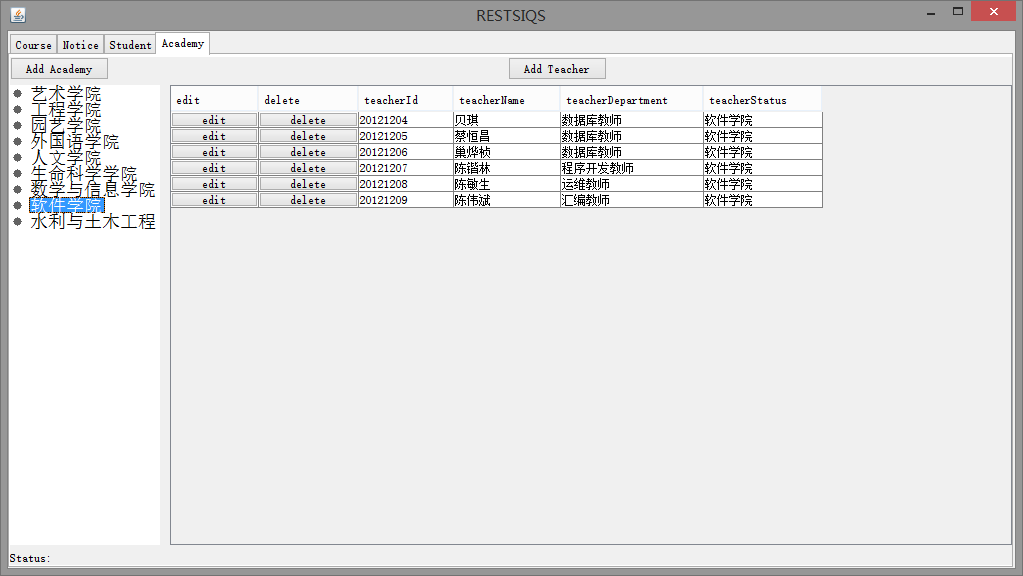


图14 学院管理界面

学院信息管理采用双层列表的形式，显示出学校的学院及其下属的教师。右击学院列表中的学院，可以管理学院级的信息，包括学院id，学院名等，同时右击学院名可以进行删除操作；单击学院名，会在右侧显示其下属的教师列表，双击教师列表可以管理教师个人信息，这方面功能可以扩展为不同教师根据自身账户登入数据管理软件，从而进行其课程下对应学生的成绩更新。

### 安卓客户端

1. 登录

系统用户使用远程数据库保存的账户密码进行登入，首次登入后会自动更新用户相关信息。如果账号密码输入有误，会显示对应的错误信息；若缺少账号密码，会要求用户输入；若手机连接远程数据库失败，会显示联网失败的提示信息。界面支持第三方输入法，同时当用户点击登入时，登入按钮会转变成不可再次点击的状态，并使用toast即时更新loading或者登入成功的系统状态。若登入失败，同样会使用toast显示登入失败的原因，包括账号密码为空、账号密码错误和网络连接失败。

用户成功登入后，会首先显示公告页面并用toast显示欢迎语句，欢迎语句中包含了所登入账户的信息，并在三秒后自动消失。



图15 安卓端登入界面

1. 更新信息

用户在登入时会自动进行一次数据更新，同时用户可以选择手动更新，更新内容包括远程数据库变更的课程信息、技能等级考试和学院公告。以远程数据库为基准值，手机端的数据只会落后于数据库或者与之同步。更新成功后，界面会显示新的数据，同时后台会即时保存数据，用户可以在断开网络后继续查看已更新的数据。

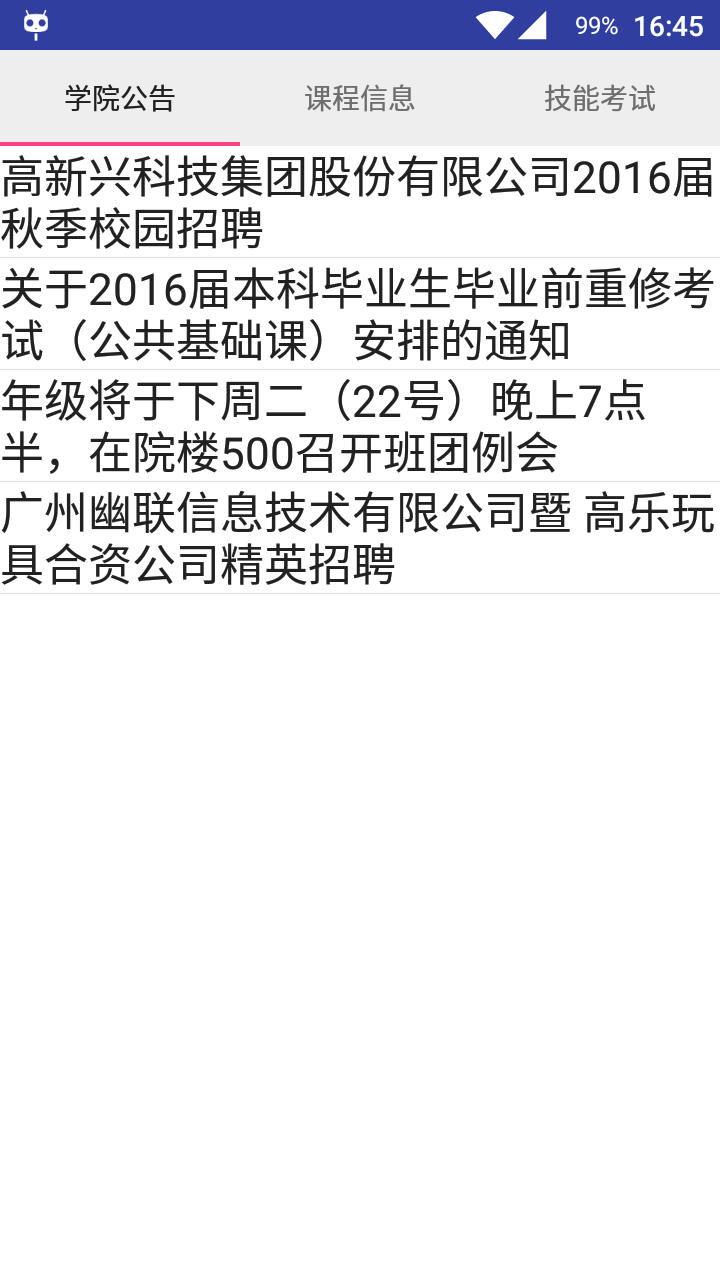


图16 更新信息界面（更新前）

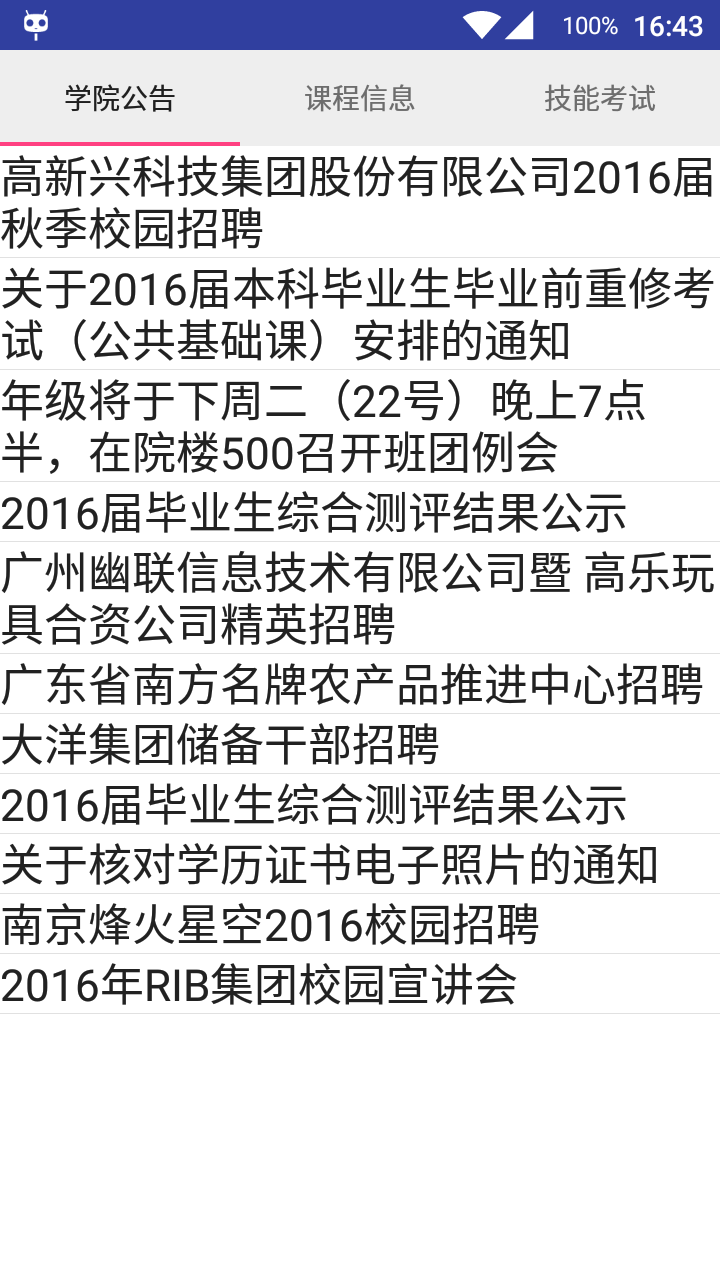


图17 更新信息界面（更新后）

1. 查看公告

在查看学院公告列表时，单击一条公告，会显示对应的详细内容。公告列表会以公告标题显示公告内容，点开公告后，界面上方显示公告标题，下方显示公告详细内容。公告界面支持滑动条，当显示内容过长的公告时会自动显示滑动条，用户可以使用常用的操作习惯来查看公告内容。

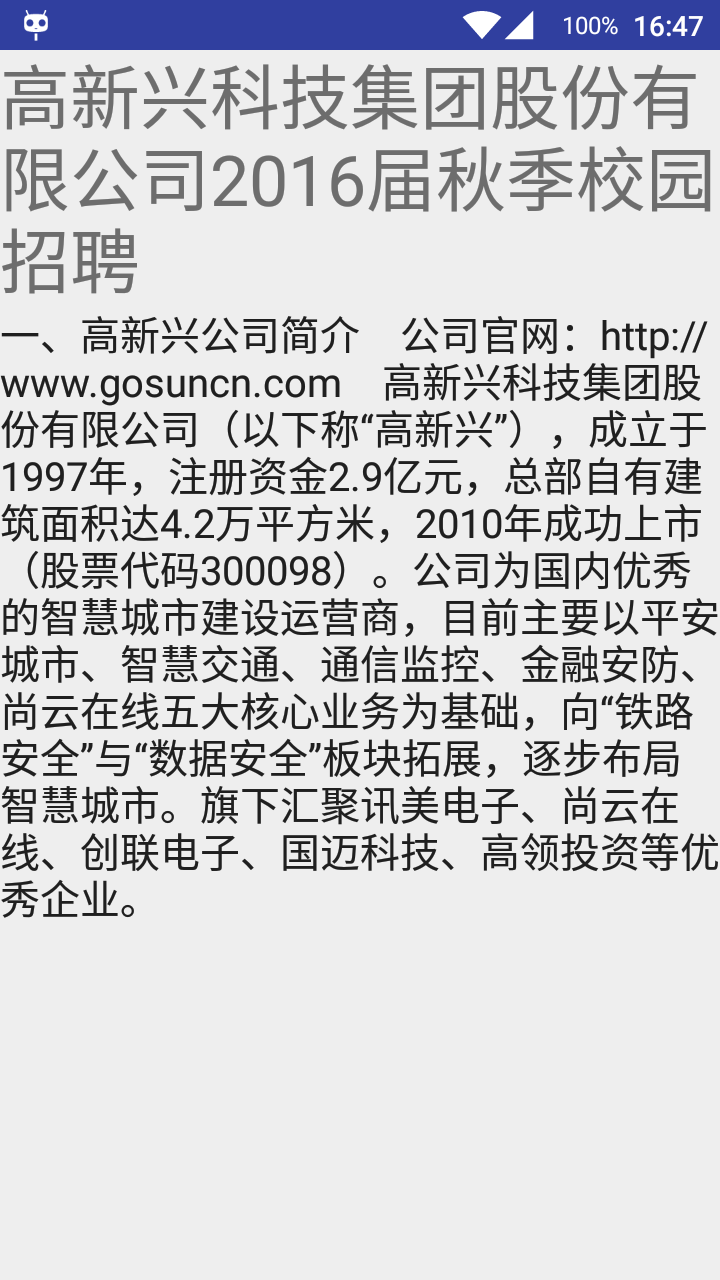


图18详细公告信息界面

1. 查看课程信息

在查看个人课程信息时，单击一项课程，会显示课程当前信息。校方会在学生完成考试后更新数据库，学生可以根据系统查看本人考试成绩。根据实际使用场景，会出现学生已经进入该课程但成绩为默认值0的情况，这是因为教师尚未更新该课程的成绩，并非系统错误。具体显示内容为：课程名称、学分、教师id、上课时间、五次练习题分数、三次测试题分数、平时分以及期末成绩，并自动为学生计算课程总分。

1. 退出

软件提供多种退出途径：安卓系统通用的退出操作、打开菜单选择退出和双击返回键退出。其中双击退出具体操作为：点击第一次返回键，界面使用toast显示退出指导，提醒并指导用户当前操作为退出操作以及具体方法，同时软件判断用户输入，若用户在一点五秒内点击第二次返回键，程序会选择退出，若用户两次操作时间超过规定秒数，程序不会退出。这能够在一定程度上防止用户误操作。

# 总结与展望

## 总结

学生信息管理系统开始设计到实现、内测历时半年，经过多次版本迭代，从搭建完整的RESTful后台，到数据管理系统实现各种数据的远程管理，以及安卓端两个大版本的更新迭代，亲身经历了一个解决方案从需求到设计到实现的艰难过程。本文完成的工作与解决的问题如下：

介绍了REST技术、SpringMVC、HttpClient、Android Support Design Library。

实现学生使用手机查看个人课程信息、所在学院发布公告的需求，并利用离线存储数据解决手机网络不稳定所导致不能随时随地查看信息的问题，减轻了服务端的网络压力。

实现远程管理阿里云数据库的数据管理软件，能够使用软件发布学院公告、管理学生账户、更新学生课程信息、管理学院教师信息的功能。

解释了学院更新学生成绩、发布公告、管理学生账户等关键技术的实现思路、操作流程以及关键代码。

## 展望

系统基本实现了学生信息管理系统的主要流程，包括管理学生账户、更新学生课程信息、发布公告，系统基本具备可运行性、可扩展性、可靠性。但是由于本人实际商用的项目经验不多，对本系统实际使用存在以下问题：

数据库只使用了单个服务和数据库，一旦用户增加，很容易导致系统崩溃。考虑到实际应用中主要资源会消耗在学生查询中，系统应该采取读写分离的模式，一方面设置主从两个数据库和服务，分别处理数据库读和写的操作，主数据库为数据基准，供数据管理者使用；从数据库定期向主数据库读取数据，主要向查询信息的学生提供读取数据库的服务。

数据管理软件应该提供足够的安全机制以及同步机制，考虑到用户增加的情况，可能会有多个教师同时更新学生成绩，这里就需要为教师提供个人账户，一方面使数据更加安全，一方面可以记录甚至回滚更新数据的各种操作。

再有就是由于数据管理者的输入界面并不够友好，这里应该增加导入excel到数据库的功能。教师可以先写好或者从其他系统获取学生成绩的excel表格，导入系统，直接完成学生成绩的更新。

参 考 文 献

韩陆.Java RESTful Web Service实战[M] .机械工业出版社，2015:79-79

Leonard Ricbardson.RESTful Web APIs[M].电子工业出版社，2014:33-33

许文稼，赵英凯. 应用SpringMVC与Hibernate进行WEB开发[J]. 计算机应用与软件，2008，25(2):264-265

舒礼莲. 基于 Spring MVC的Web应用开发[A]. 计算机与现代化，2013，219(11):167-173

王超，闾陈莉，吴迪，项英杰. 基于HttpClient的Android客户端的设计与实现[A]. Computer Era，2014，30(3):30-32

Pongsathorn Noimanee，Yachai Limpiyakorn.Towards a RESTful Process of Conference Management System[J]. Proceedings of the International MultiConference of Engineers and Computer Scientists，2009，3(1):18-20

Praveen Gupta，Prof. M.C. Govil. Spring Web MVC Framework for rapid open source J2EE application development: a case study[J]. International Journal of Engineering Science and Technology，2010，2(6): 1684-1689

C. Bruce Kavan，Margaret T. O’Hara，Edward C. Patterson，Robert P. Bostrom. Excellence in client/server information system implementations: understanding the STS connection[J].Management Decision，1999，37(3):295-301

Julia Janicki，Nitish Narula，Matt Ziegler，Benoit Guénard，Evan P. Economo. Visualizing and interacting with large-volume biodiversity data using client–server web-mapping applications: The design and implementation of antmaps.org[J]. Ecological Informatics，2016，32(1)：185-193

Mou’ath Hourani，Qusai Shambour， Ahmad Al-Zubidy，Ali Al-Smadi. Proposed Design and Implementation for RESTful Web Server[J]. JOURNAL OF SOFTWARE，2014，9(5):1071-1080

致 谢

在论文即将完成之际，感谢半年来对我关心与帮助我的指导老师张义青老师，在张老师悉心的指导下，顺利完成的系统的设计、实现以及论文的撰写。

在系统的设计与实现过程中，从需求分析、功能模块划分、流程图制作、系统架构的设计，让我更深刻地理解软件工程；从实体图的绘制、表结构的设计，让我更深刻地意识到数据库设计的重要；从代码架构的搭建，具体业务逻辑的实现，让我更深刻地感受到代码可重用的魅力。

大学的四年里，感谢所有的老师给予我们的知识与锻炼，感谢学院对我专业技能的培养，才让我有机会进入IT行业的大时代。最后，感谢华南农业大学。

**华南农业大学**

**本科生毕业论文成绩评定表**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 学号 | 201231000826 | | | | | 姓名 | | 全程 | 专业 | 软件工程 | | |
| 毕业论文题目 | | | 基于Restful方式的在线学生信息管理系统 | | | | | | | | | |
| 成绩（百分制）：　　　　　　　　　　　　　　　　　　　　指导教师签名：　　　　　　　　　　年　　月　　日 | | | | | | | | | | | | |
| 评  阅  人  评  语  及  成  绩  评  定 | | 成绩  评定  标准 | | **评分项目** | | | | | | | **分值** | **得分** |
| 选题  质量  20% | 1 | | 专业培养目标 | | | | 5 |  |
| 2 | | 课题难易度与工作量 | | | | 10 |  |
| 3 | | 理论意义或生产实践意义 | | | | 5 |  |
| 能力  水平40% | 4 | | 查阅文献资料与综合运用知识能力 | | | | 10 |  |
| 5 | | 研究方案的设计能力 | | | | 10 |  |
| 6 | | 研究方法和手段的运用能力 | | | | 10 |  |
| 7 | | 外文应用能力 | | | | 10 |  |
| 成果  质量40% | 8 | | 写作水平与写作规范 | | | | 20 |  |
| 9 | | 研究结果的理论或实际应用价值 | | | | 20 |  |
| 评阅人评语：  成绩（百分制）：　　　　　　　　　　　　　评阅人签名：　　　　　　　　　　　　年　　月　　日 | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 答  辩委员会  意见与  成  绩  评  定 | 评价项目 | 具体要求（A级标准） | 最高分 | 评分 | | | | |
| A | B | C | D | E |
| 论文质量 | 论文（设计）结构严谨，逻辑性强；有一定的学术价值或实用价值；文字表达准确流畅；论文格式规范；图表（或图纸）规范、符合要求。 | 60 | 55-60 | 49-54 | 43-48 | 37-42 | ≤36 |
|  |  |  |  |  |
| 论文报告、讲解 | 思路清晰；概念清楚，重点（创新点）突出；语言表达准确；报告时间、节奏掌握好。 | 20 | 19-20 | 17-18 | 15-16 | 13-14 | ≤12 |
|  |  |  |  |  |
| 答辩情况 | 答辩态度认真，能准确回答问题 | 20 | 19-20 | 17-18 | 15-16 | 13-14 | ≤12 |
|  |  |  |  |  |
| 是否同意通过论文答辩（打√）   1. 同意 2. 不同意   成绩（百分制）：　　　　　　　　 答辩委员会主席（签名）：  　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　 年 月 日 | | | | | | | |

续上表：

**注：1、**论文成绩评定等级：参考评阅人的评阅、指导教师评阅情况，结合答辩情况，建议按指导教师评分、评阅人评分、答辩评分为4∶3∶3的比例评定论文总成绩分数，然后按优（90－100分）、良（80－89分）、中（70－79分）、及格（60－69分）、不及格（<60分）给出成绩等级。**2**、论文成绩以分数的形式登记到教务管理系统。