

新文献检测报告（简明版）

报告编号：PL-20200517-C13EE4B1-JM

检测时间：2020-05-17 18:06:41

题名：基于Android的掌上校园APP开发论文

作者：佚名

检测范围：☒中国学术期刊数据库 ☒中国学位论文全文数据库 ☒中国学术会议论文数据库
☒中国学术网页数据库 ☒中国专利文献数据库 ☒中国优秀报纸数据库

检测结果

% 总相似比：2.51%

检测字数：17318

参考文献相似比：0.14% / 排除参考文献相似比：2.36%
可能引用本人已发表论文相似比：0.00% / 辅助排除本人已发表论文相似比：2.51%
可能引用本人学位论文相似比：0.00% / 辅助排除本人学位论文相似比：2.51%
单篇论文最大相似比：0.47%

相似片段分布图



注：绿色区域为参考文献相似部分，蓝色区域为本人已发表论文相似部分，黄色区域为本人学位论文相似部分，红色区域为其他文献相似部分

相似文献列表

| 序号 | 相似比 | 题名 | 作者 | 文献类型 | 来源 | 发表时间 | 是否引用 |
|----|-------|--------------------------|--------------|------|--|------------|------|
| 1 | 0.47% | 基于Android的校园通掌上系统的设计与实现 | 张达 | 学位论文 | 天津大学 | 2015-10-01 | 否 |
| 2 | 0.36% | 建立健全博士研究生培养创新机制 | 曾家刚 等 | 期刊论文 | 《西南交通大学学报（社会科学版）》 | 2004-06-14 | 否 |
| 3 | 0.22% | 认知风格与样例的呈现方式对数学学习的影响研究 | 郝婷 | 学位论文 | 浙江师范大学 | 2010-04-10 | 否 |
| 4 | 0.17% | 基于CMS的铁路客网站管理系统的设计与实现 | 邹永平 | 学位论文 | 北京交通大学 | 2014-06-01 | 否 |
| 5 | 0.14% | 基于安卓平台的掌上校园系统的设计与实现 | 闫素旺 | 学位论文 | 河北工业大学 | 2013-05-01 | 是 |
| 6 | 0.12% | 基于云存储的污水处理数据集中监控系统的研究与实现 | 王永振 | 学位论文 | 桂林理工大学 | 2018-05-01 | 否 |
| 7 | 0.11% | 高速列车牵引运行仿真及虚拟现实的设计与实现 | 李阳 | 学位论文 | 东北大学 | 2013-07-01 | 否 |
| 8 | 0.11% | 基于本体的资源管理系统设计与实现 | 童吕萍 | 学位论文 | 湖南大学 | 2010-04-22 | 否 |
| 9 | 0.10% | 基于SIP协议的实时语音通信系统的研究与开发 | 罗丛优 | 学位论文 | 暨南大学 | 2016-06-02 | 否 |
| 10 | 0.10% | 分布式异构农业数据库集成中间件系统 | 尚明华 等 | 会议论文 | 中国农科院农业信息研究所建所50周年庆祝大会暨中国农业信息科技创新与学科发展大会 | 2007-10-19 | 否 |
| 11 | 0.10% | 201510345408.7 | 苏宁云商集团股份有限公司 | 专利论文 | 中文专利全文数据库 | 2015-06-17 | 否 |
| 12 | 0.09% | 北京医保数据采集子系统的开发与研究 | 胡志平 | 学位论文 | 北京工业大学 | 2008-04-01 | 否 |
| 13 | 0.09% | 基于Android的餐厅点餐系统的设计与实现 | 胡向颖 | 学位论文 | 青岛科技大学 | 2018-04-26 | 否 |
| 14 | 0.09% | 基于VFP6的学籍成绩管理信息系统的设计与实现 | 江世军 | 学位论文 | 电子科技大学 | 2010-12-01 | 否 |

| | | | | | | | |
|----|-------|----------------------|------|------|------------|------------|---|
| 15 | 0.08% | 基于WBS的工程项目管理系统的设计与实现 | 张守领 | 学位论文 | 电子科技大学 | 2013-12-24 | 否 |
| 16 | 0.08% | 面向大规模定制的客户需求信息系统研究 | 岳同启 | 学位论文 | 大连理工大学 | 2004-03-01 | 否 |
| 17 | 0.08% | 利用移动互联网技术构建实时交互学习平台 | 柯伟 等 | 期刊论文 | 《电气电子教学学报》 | 2014-12-28 | 否 |

 [原文](#)

成绩

本科毕业论文（设计）

题目： 基于Android的掌上校园APP开发

学生姓名 王琨鹏
 学 号 2016118117
 指导教师 董乐红
 院 系 软件学院
 专 业 软件工程
 年 级 2016



诚信声明

本人郑重声明：本人所呈交的毕业论文（设计），是在导师的指导下独立进行研究所取得的成果。毕业论文（设计）中凡引用他人已经发表或未发表的成果、数据、观点等，均已明确注明出处。除文中已经注明引用的内容外，不包含任何其他人或集体已经发表或在网上发表的论文。

特此声明。

论文作者签名： 王琨鹏

日 期： 2020年3月21日

摘要

在这个信息科技技术飞速发展的时代，人们的生活和网络已经密不可分，互联网的快捷与便利也为人们提供了更多更快获取信息的渠道和方法，在高校校园这个以学生为主要组成部分的社会场所，通常信息的访问形式主要是以学生的需求为主，针对这一需求，一款可以供学生查阅信息，了解校园资讯的APP作为一种新的信息查阅传播方式为广大学生提供了一种便捷的途径去获取信息。

掌上校园APP是一款Android平台的手机APP，由客户端和服务端，以及网络传输线路组成，本文介绍的系统目的在于在Android平台上为学生提供信息查阅功能，系统结构采用了C/S结构，使用基于Unity的GUI布局来实现UI的布局结构，由C#语言开发，本系统提供了学生成绩查询，课程查询，校园资讯的浏览，调查问卷的填写四个功能模块，满足学生对信息查询的需求。

本系统在PC端上搭建Android环境模拟手机平台正常运行，使用简单，界面简洁，在一定程度上可以提高学生获取信息和学习的效率。

关键字：Unity 安卓 客户端 服务端



Abstract

In this era of rapid development of information technology, people's lives and networks are inseparable. The speed and convenience of the Internet also provide people with more channels and methods for faster and faster access to information. In the college campus, students are the main component. In the social place, usually the access form of information is mainly based on the needs of students. In response to this demand, an APP that can be used by students to consult information and understand campus information is a new way of information dissemination and communication for students.

The handheld campus APP is an Android platform mobile phone application, which is composed of a client and a server, and a network transmission line. The purpose of the system introduced in this article is to provide students with information retrieval function on the Android platform. The system structure uses a C / S structure , Using the Unity-based GUI layout to achieve the UI layout structure, developed by C # language, this system provides four functional modules for student score query, course query, campus information browsing, and questionnaire filling, to meet the students ' information query requirements demand.

This system builds an Android environment on the PC to simulate the normal operation of the mobile phone platform. It is simple to use and has a simple interface. To a certain extent, it can improve the efficiency of students in obtaining information and learning.

Keywords: Android; Unity; Servers; Cliend;

目 录

第一章 序言8

1.1 课题背景及概述8

1.2 课题的研究目的及贡献8

1.3 课题内容叙述及本文安排8

第二章 掌上校园系统的相关技术10

2.1 Android的发展及简介10

2.2 C/S架构的简要分析10

2.3 MySQL数据库11

2.4 UDP协议及Socket流程11

2.5 Unity GUI工具12

第三章 系统功能介绍13

3.1 需求分析13

3.2 客户端功能介绍13

3.2.1 客户端使用流程13

3.2.2 用户注册14

3.2.3 用户登录15

3.2.4 成绩查询17

3.2.5 课程查询18

3.2.6 校园资讯19

3.2.7 调查问卷功能20

第四章 数据库设计21

4.1 数据库结构21

4.2 数据库逻辑结构设计21

4.3 数据库表23

| | |
|----------------|----|
| 表二 课程信息表 | 24 |
| 表三 校园咨询表 | 24 |
| 表四 调查问卷表 | 24 |
| 第五章 模块交互与设计 | 26 |
| 5.1 客户端服务端通信 | 26 |
| 5.1.1 客户端发送流程 | 26 |
| 5.1.2 服务端接收流程 | 26 |
| 5.2 服务端数据库读取数据 | 27 |
| 5.3 本地数据存取 | 28 |
| 5.4 事件逻辑 | 29 |
| 5.4.1 登录事件 | 29 |
| 5.4.2 注册事件 | 30 |
| 5.4.3 界面切换 | 31 |
| 5.4 界面数据显示 | 32 |
| 5.5 数据请求逻辑 | 33 |
| 5.5.1 初始化逻辑 | 33 |
| 5.5.2 数据查询 | 35 |
| 5.5.3 数据插入 | 35 |
| 5.5.4 执行数据库语句 | 36 |
| 5.5.5 数据库关闭 | 37 |
| 第六章 用户界面 | 39 |
| 6.1 注册登录界面 | 39 |
| 6.2 菜单界面 | 40 |
| 6.3 成绩查询界面 | 40 |
| 6.4 课程查询界面 | 41 |
| 6.5 校园资讯查看界面 | 42 |
| 6.6 调查问卷界面 | 43 |
| 第七章 总结及反思 | 45 |
| 参考文献 | 47 |



序言

课题背景及概述

随着互联网技术的越来越成熟，以及4G网络的大面积覆盖，移动手机的普及使得人与人之间的沟通变得简单而高效。历经三十多年的发展，在拥有了众多的互联网用户之后，我国网络用户的主力军学生群体占到的比重是非常大的，所以一款为学生提供服务，便利学生需求的软件就应运而生，在日常的学习生活中，学生群体在对自己的个人信息查询以及校园的相关信息方面有很多的访问需求，而本文的研究课题，基于Android的掌上校园APP就是这么一款针对学生群体的软件，为了方便学生群体的日常学习生活，有效的提高学生的学习效率，本课题的研究系统会对针对学生群体的需求，对学生最基本的信息需求做以支持。

课题的研究目的及贡献

本课题旨在实现一个基于Android平台的掌上校园APP，提供Android平台上学生查阅个人信息，浏览学校信息的功能，方便学生对个人以及学校情况的及时了解和掌握，学生可以在客户端登录自己注册的账号来查看信息，这保证了个人信息的安全性。本系统提高的功能可以在一些方面提高学生的学习效率和掌握信息的便捷性。

课题内容叙述及本文安排

2

在本文中，将对课题研究分析的课题从以下几个方面开始叙述，第二章将从系统的技术支持方面做以叙述，讲述实现本系统所需要的技术点，第三章将叙述系统的需求分析以及包含的功能。第四章主要讲述的内容是服务端的数据库分析与设计，第五章主要是叙述系统之间各个模块是如何交互和通信流程，第六章则会去描述系统的一些重要组成界面，最后为本系统实现完成后的总结。

3



掌上校园系统的相关技术

2.1 Android的发展及简介

Android是一个以Linux为基础的操作系统，主要用于移动设备，它大致分为四层架构，Linux内核层，系统运行库层，应用框架层和应用层。第一层Linux层主要是为Android系统提供硬件上的支持，第二层，系统运行库层，它提供Android应用运行时的必要的核心库，比如图形渲染，数据库，应用程序运行需要的虚拟机等等，第三层，应用框架层，提供Android的API(应用程序接口)，供Android应用或者开发者使用，第四层，应用层，包含Android系统安装的手机应用程序(联系人，记事本等等)。经过多年的发展，安卓以及发展成全球第一大操作系统，本次的掌上校园系统也是基于Android平台开发的。

2.2 C/S架构的简要分析

C/S架构是一种软件架构，主要应用于局域网内，它包含的最流行的两层架构结构示意图如图2.2所示：

图2.2 C/S两层架构图这是最常用的两层结构，本次的系统也将采用两层的结构来实现，两层结构分为4客户端和服务端，客户端主要是业务逻辑和表现层的显示，处理一些数据的存储

发送和数据的计算等任务，它直接与服务器相连，服务器通常用来处理事务逻辑，客户端的通信和数据库的增删改查等重要功能。

本系统采用C/S架构来实现学生客户端和用来存储数据的客户端之间的交互，主要是由于C/S架构的以下几个优点：C/S架构是客户端和服务端的点对点通信，使得通信和数据传输更加安全，在流量和网络通信上性能较好，客户端也可以处理一些逻辑事务，数据计算等复杂的任务，有助于减轻服务端的压力。

2.3 MySQL数据库

MySQL是一个关系型数据库，基于SQL语言进行增删改查，它是一种用来存储数据的应用程序，MySQL是一个轻量级的数据库系统，轻量易于维护的特性也是我选择用它来实现本系统的原因之一，它拥有完备的数据操作功能，满足大部分系统需要的对数据操作的需求，线程的特性使它可以灵活的为用户提供多种服务，而且MySQL作为一种开源的应用程序，用户可以根据自己的项目需求对它进行相应的修改，良好的可扩展性使它在很大程度上很受欢迎。

2.4 UDP协议及Socket流程

UDP协议是一种网络传输协议，它是无连接的不可靠协议，即UDP协议不保证网络传输中的数据包是否可以准确的到达目标计算机，在使用UDP协议进行网路数据传输时，无需发送方和接收方建立连接，所以相对TCP协议UDP比较节省时间，由于省略了建立连接的过程，所以UDP协议的数据传输效率是高于TCP协议的，但是它的传输准确性是无法得到保障的，但是UDP协议的通信流程相对简单高效，且维护成本较低，所以常用作局域网中C/S架构的应用程序。

5

Socket通信俗称套接字，它是一种网络通信的抽象层，应用程序可以通过它在连接建立之后直接进行通信和数据的传输，既可以由服务端广播消息，也可以

由客户端发送消息，通信的灵活性较高，Socket通信具有传输稳定，丢包率小，而且传输方式简单可靠，具有很好的可扩展性，它的通信流程如图2.4所示：

图2.4 Socket通信流程图

基于UDP协议数据传输的简单高效，本系统将采用UDP协议的Socket通信流程来进行客户端和服务端的数据传输。

2.5 Unity GUI工具

UGUI系统是基于Unity的用于制作UI界面的工具，具有体积小，快速和可视化的优点，在开发过程中，可以对开发效率有很大的提升，易于使用和扩展，可以再写少量代码的情况下去实现一个项目的大部分UI界面。

6

系统功能介绍

3.1 需求分析

1. 注册登录功能
2. 个人成绩查询功能
3. 个人课程信息查询功能
4. 查看校园相关资讯，新闻等功能
5. 填写调查问卷并提交的功能
6. 退出APP

3.2 客户端功能介绍

3.2.1 客户端使用流程

本系统是针对在校学生的一款查询信息的软件，主要包含的功能模块有个人成绩查询、课程信息查询、校园资讯查看、提交调查问卷功能。用户进入本系统后，可以点击界面的四个按钮去进行操作，完成自己需要的信息查询功能。

7

系统功能框架如图3.2.1所示：

图3.2.1 系统框架图

3.2.2 用户注册

此用例用于实现用户进行账户注册的功能，用例描述如下所示

用例名称 账户注册

用例说明 主事件流：

用户点击APP进入应用，用例开始

系统显示注册界面

用户填写账户名称

用户填写密码

用户点击注册按钮

8

系统提示结果(注册成功/注册失败)

用户注册流程图如图3.2.2所示：

图3.2.2 注册用例图

3.2.3 用户登录

此用例用于实现用户进行账户登录的功能，用例描述如下所示

用例名称 用户登录

用例说明 主事件流：

9

用户点击APP进入应用，用例开始

系统显示登录界面

用户填写账户名称

用户填写密码

用户点击登录按钮

系统提示结果(登录成功/登录失败)

用户登录流程图如3.2.3图所示：

图3.2.3 登录用例图

10

3.2.4 成绩查询

此用例用于实现用户进行成绩查询的功能，用例描述如下所示

用例名称 成绩查询

用例说明 主事件流：

用户点击基础功能按钮，用例开始

系统显示基础功能列表

用户点击成绩查询

系统跳转至成绩查询界面

用户选择学年

用户查询按钮

系统展示查询结果

用户成绩查询流程图如3.2.4图所示：

图3.2.4 成绩查询用例图

11

3.2.5 课程查询

此用例用于实现用户进行课程查询的功能，用例描述如下所示

用例名称 课程查询

用例说明 主事件流：

用户点击基础功能按钮，用例开始

系统显示基础功能列表

用户点击课程查询

系统跳转至课程查询界面

用户选择学年

用户查询按钮

系统展示查询结果

用户课程查询流程图如3.2.5图所示：

图3.2.5 课程查询用例图

12

3.2.6 校园资讯

此用例用于实现用户进行校园资讯查阅的功能，用例描述如下所示

用例名称 校园资讯查阅

用例说明 主事件流：

用户点击基础功能按钮，用例开始

系统显示基础功能列表

用户点击校园资讯

系统跳转至校园资讯界面

系统展示查询结果

用户查看资讯用例图如3.2.6图所示：

图3.2.6 查看资讯用例图

13

3.2.7 调查问卷功能

此用例用于实现用户进行提交调查问卷的功能，用例描述如下所示

用例名称 提交调查问卷

用例说明 主事件流：

用户点击基础功能按钮，用例开始

系统显示基础功能列表

用户点击调查问卷

系统跳转至校园资讯界面

用户填写调查问卷详细信息

用户点击提交按钮

系统展示提交结果

用户注册流程图如3.2.7图所示：

图3.2.7 调查问卷用例图

14

数据库设计

4.1 数据库结构

本系统使用的**数据库**软件是**MySQL数据库**，第一步需要在服务器创建数据库，随后在数据库添加数据表，通过数据表来存储数据以供用户访问，具体的操作步骤如下：

打开MySQLWorkBench应用程序，右键选择创建数据库并为数据库命名

点开创建的数据库，即可选择创建数据表，视图等

点击创建表(Create Table)

编辑创建的数据表并设置表的数据项和属性值，其中Table Name为表名，DataType为数据类型，PK为主键，NN为值是否可以为空，UQ代表外键，AI表示数据自动扩增等

随后点击应用保存属性和数据项

这就是通过MySQL创建一个数据库和数据表的方法，接下来通过这个方法可以创建多个数据表用来存储系统需要保存的数据，根据本系统的实际情况，需要创建的表有以下几个：

users(用来存储用户信息)

courses(用来存储课程信息)

news(用来存储校园资讯数据)

questionnaires(用来存储提交的调查问卷数据)

4.2 数据库逻辑结构设计

通过分析本系统的用户需求和系统功能需求，可以对数据库的设计做以详

15

细的叙述和整理，结合上一小节对数据库的概要设计，对掌上校园系统的数据库详细设计如下：

学生用户的数据表包含了学生姓名，学生学号，用户名，密码，入学年份，性别这些必要属性，其中学号为主键，作为区分用户的标识，学生数据表的结构如图4.2所示：

图4.2.1 学生信息结构图

课程信息表由课程ID，课程名，学分，开设学院，年级属性组成，其中课程ID作为主键，用来区分不同课程，课程信息表的数据表结构如图4.2.2所示：

图4. 2. 2 课程信息结构图

16

校园资讯表由资讯ID，校园资讯日期，标签，发布者，资讯内容组成，其中资讯ID作为主键，校园资讯表的数据表结构如图4. 2. 3所示：

图4. 2. 3 校园资讯结构图

调查问卷表由问卷ID，问卷发布日期，发布者，问卷内容，提交者组成，其中问卷ID作为主键，校园资讯表的数据表结构如图4. 2. 4所示：

图4. 2. 4 调查问卷结构图

4. 3 数据库表

17

表1 学生信息表

字段名
数据类型
长度
主键
备注

Id
Int
15
是
自动生成

Sex
Varchar
10
否
性别

Account



Varchar

50

否

账户名

Name

Varchar

50

否

用户密码

Password

Varchar

20

否

学生姓名

Year

Int

10

否

入学年份



表二 课程信息表

字段名

数据类型

长度

主键

备注

Id

Int

15

是

自动生成

Grade

Varchar

10
否
年级

Name
Varchar
50
否
课程名

Score
Int
10
否
学分

Departments
Varchar
50
否
院系

表三 校园咨询表

字段名
数据类型
长度
主键
备注

Id
Int
15
是
自动生成

Date
Varchar
20
否



日期

Tags

Varchar

50

否

标签

Announcer

Varchar

20

否

发布者

Details

Varchar

1000

否

资讯内容

表四 调查问卷表

18

字段名

数据类型

长度

主键

备注

Id

Int

15

是

自动生成

Date

Varchar

20

否

日期

Submitter

Varchar

20

否

提交者

Announcer

Varchar

20

否

发布者

Details

Varchar

1000

否

问卷内容



19

模块交互与设计

客户端服务端通信

客户端发送流程

本系统使用的是基于UDP协议的Socket通信来实现客户端和服务端的通信，C#语言提供了一个封装好的Socket类，在使用它时只需要按照Socket通信的流程即可收发数据。

第一步，创建客户端Socket对象，

例：`Socket udpClient = new Socket(AddressFamily.InterNetwork, SocketType.Dgram, ProtocolType.Udp);`其中，第一个参数为通信的地址族，常用的为IPV4地址族，第二个参数为Socket对象的类型，第三个参数为通信的协议，本次使用的协议为UDP协议。

第二步，读取本机IP地址和端口号，例：`IPAddress.Parse("192.168.1.23")`，使用该函数可以获取本地的IP地址。

第三步，绑定本地IP地址和端口号，代码如下：

`EndPoint remoteEndPoint = new IPEndPoint(ipaddress, 7788);`绑定好IP地址和端口号后，即可通过send方法发送数据，方法如下：

`udpClient.SendTo(data, remoteEndPoint);`此方法代表的意义为将数据发送至IP地址为remoteEndPoint，即上述的IP地址，上面三步即可完成客户端的数据发送任务。

5.1.2 服务端接收流程

20

服务端接收数据的方法和客户端类似，第一步和第二步一致，都需要创建Socket对象和绑定IP地址和端口号，再完成上述步骤之后，第三步需要接收数据。

例：`EndPoint remoteEndPoint = new IPEndPoint(IPAddress.Any, 0);`

udpServer.ReceiveFrom(data, ref remoteEndPoint);使用这两句代码方法可以接收到来自任意IP地址发送到本机的数据，因为使用的是UDP协议的缘故，所以本系统需要开启一个线程来不断的接收数据。

服务端数据库读取数据

服务端读取MySQL数据库的数据首先需要在项目中引入四个动态链接库，链接库分别为：

I18N.CJK.dll

I18N.dll

I18N.West.dll

MySql.Data.dll

引入四个链接库后，在服务端的命名空间中引入三个类库如下：

```
using System.Data;
```

```
using MySql.Data.MySqlClient;
```

```
using System.Diagnostics;
```

在完成上述两个步骤之后，服务器便可以读取本机MySQL数据库的数据了，具体的读取方法如下：

第一步，设置连接类对象，static MySqlConnection mySqlConnection;该对象用于之后链接数据库的主要操作者。

第二步，为该对象设置连接具体需要连接的数据库对象，mySqlConnection = new

21

MySqlConnection(mySqlString);该方法可以连接到mySqlString对应信息的数据库。其中mySqlString的值应为如下的信息：

mySqlString = "Database=" + databaseName + ";DataSource=" + host + ";User=" + userName + ";Password=" + password + ";port=" + port + ";";该字符串类型的数据明确指定了需要连接的数据库名，数据资源，数据库用户名和密码(用于验证身份)，以及端口号，设置完成后，便可以连接并访问数据库的数据表。

第三步，读取数据库的数据表，方法示例：

MySqlDataAdapter mySqlAdapter = new MySqlDataAdapter(sqlString, mySqlConnection);该方法的第一个参数为数据库的操作语句，第二个参数为我们先前设置好的连接对象，接下来通过mySqlAdapter.Fill(ds);将获得的数据填充到ds对象中，其中ds对象为一个C#提供的DataSet类型的对象，填充完成后访问该对象即可获得我们查询的数据。

本地数据存取

除了需要从服务端读取数据外，我们还需要处理本地数据的读取，因为有的数据例如用户选择保存的账户名和密码，之前查询的数据的缓存等等，都可以保存在本地，再次使用时从本地加载即可，这样可以提升整个系统的响应速度，给使用者带来较好的体验，读取和保存数据的方法本系统使用的是JSON格式文件的存取，实现方法如下：

第一步，需要在项目工程引入 Newtonsoft.Json.dll动态链接库，然后在项目的命名空间引入库文件，using Newtonsoft.Json;执行这两步操作即可通过JSON的方法存取数据了。

第二步，jsonStr = JsonConvert.SerializeObject(save);使用该方法将数据转换为json字符串进行保存，保存方法如下：
StreamWriter writer = new StreamWriter(filePath);

22

```
writer.Write(jsonStr);
```

```
writer.Close();
```

通过上述三句代码可以将json数据存储到本地json文件中，读取的方法也和保存的方法类似，先将本地json文件读取出来

，如下所示：

```
StreamReader reader=new StreamReader(filePath);  
string jsonStr = reader.ReadToEnd();  
reader.Close();
```

之后将读取到的json字符串转化为所需的数据对象，方法如下：

```
JsonConvert.DeserializeObject<Save>(jsonStr);
```

通过该方法得到数据对象后便可以对数据对象进行操作，例如加载到UI界面等等。

5.4 事件逻辑

5.4.1 登录事件

当用户点击界面的登录按钮时，系统会去从数据库读取账号和密码的信息并返回与输入文本框的数据做对比，若数据验证正确则显示登录成功后的界面，具体的实现逻辑如下：

第一步:先判定用户名和密码是否输入作为安全校验，若任一为空则不进行后续的业务流程。

第二步:通过安全校验后，按照用户名来查询数据库的密码是否和用户输入的一致，代码：

```
mysql.OpenSql();连接数据库
```

```
DataSet ds = mysql.Select("users", new string[] { "user_password", "user_id" },
```

```
23
```

```
new string[] { "user_name" }, new string[] { "=" }, new string[] { userNameInput.text, });用于获取数据库的  
数据(密码)。
```

```
DataTable table = ds.Tables[0];将数据库的数据添加到一个DataTable类型的临时变量中。
```

```
foreach (DataRow row in table.Rows) {  
    foreach (DataColumn col in table.Columns) {  
        data.Add(row[col].ToString());  
    }  
}
```

以上代码片段的作用是遍历上述申请的临时变量对象，将数据添加到data数据列表中，之后通过比对data数据列表中的密码值和文本框的密码是否一致即可判定是否登录成功。

第三步:数据对比结果处理，若用户名不存在，则提示用户用户名不存在需要重新输入用户名或者去注册，若密码不正确则提示密码不正确，重新输入。

5.4.2 注册事件

当用户点击界面的注册按钮时，系统会先去进行安全校验，校验用户名和密码是否输入，之后向数据库插入用户输入的用户名和密码并返回一个bool类型的值，通过这个值可以判定插入是否成功或失败，若返回true值说明说明用户名不存在，注册成功并且可以去登录，若返回false值说明用户名存在，注册失败。具体实现逻辑如下：

第一步:安全校验，代码：

```
if (userNameInput.text == "") {  
    Message("用户名不能为空");  
}  
  
else if (passwordInput.text == "") {
```

```
        Message("密码不能为空");  
    }  
    else {  
24
```

```
        //注册逻辑  
    }
```

以上代码片段是安全校验的逻辑。

第二步:安全校验通过,进行插入数据, `bool succ = mysql.Insert("users", new[] { "user_name", "user_password"}, new[] { userNameInput.text, passwordInput.text });`通过这句代码可以返回一个succ的bool值,用来判断是否插入成功。

第三步:根据succ的值进行后续逻辑处理,代码:

```
if (succ) {  
    Message("注册成功,请登录");  
}  
else {  
    Message("注册失败,请重新注册");  
}
```

以上代码片段会给用户进行提示,表明注册操作的执行结果。

5.4.3 界面切换

用户在进行登录成功界面会切换到菜单功能界面,具体的实现逻辑如下,首先获取菜单功能界面的对象, `GameObject menu=GameObject.Find("Menu");`获取到菜单界面的对象(menu)-之后,将本来为隐藏状态的菜单界面进行显示即可,代码:`menu.SetActive(true);`若需要将原有界面隐藏掉还需要执行隐藏逻辑代码,例如功能模块的具体界面之间的切换,在成绩界面进行返回操作时,需要将成绩界面隐藏掉,实现代码为:

```
GameObject score=GameObject.Find("Score");  
score.SetActive(false);
```

上述代码片段即可将界面隐藏掉。

25

界面数据显示

在用户点击可功能按钮之后,除了需要上述的界面切换逻辑还需要将对应的数据显示到切换的目标界面,通过获取功能按钮的类型去判定需要读取哪个数据库的数据并展示到界面中,每个界面展示的具体逻辑大同小异,在此将选择成绩查询的界面做以讲述,具体实现逻辑如下:

第一步,获取功能按钮的类型,代码如下:

```
BtnType curType = this.FeatureConrroller.type;
```

在按钮的功能控制组件FeatureConrroller中获取一个BtnType类型的type属性,BtnType类型为一个枚举类型的变量,该枚举类型包含四个值,分别对应四个功能模块。获取到之后根据获取的值去执行界面数据的展示逻辑。

第二步,展示数据,例:需要展示的为成绩界面的数据,首先根据用户name从成绩数据表查询成绩数据,代码如下:

```
mysql.OpenSql();连接数据库
```

```
DataSet ds = mysql.Select("scores", new string[] { "user_name",
```

```
"user_year", "user_subject", "user_score", "user_achievement"}, new string[] { "user_name" }, new string[] {  
"=" }, new string[] { userNameInput.text, });
```

上诉代码片段会获取到一个DataSet 类型的对象, 该对象包含用户名对应的用户入学年份, 用户的科目, 用户的科目对应成绩, 用户科目对应的学分这些数据。

第三步, 将获取的数据存取到临时data数据列表中, 代码如下:

DataTable table = ds.Tables[0];将数据库的数据添加到一个DataTable类型的临时变量中。

```
foreach (DataRow row in table.Rows) {  
    foreach (DataColumn col in table.Columns) {  
        data.Add(row[col].ToString());  
    }  
}
```

26

以上代码片段的作用是遍历上述申请的临时变量对象, 将数据添加到data数据列表中, 之后将获取到的数据写入界面的对应显示位置即可, 代码如下:

```
Text[] textList= this.GetComponent<FeatureConrroller>().TextList;
```

这句代码可以获取到界面功能控制组件的TextList属性, 该属性为一个Text类型的文本数组, 之后为该文本数组依次赋值即可, 代码如下:

```
for(int i=0, i<data.Length;i++){  
    textList[i].text=data[i].ToString();  
}
```

上述代码片段遍历了获取的数据列表data并将它的数据转换成string类型赋给文本列表的text属性值, 这样就可以将成绩表查询到的数据展示在成绩界面上了。

数据请求逻辑

初始化逻辑

在初始化进行的操作主要有声明连接类对象, 绑定本地的IP地址, 端口号, 登录数据库的用户名, 密码以及需要访问的数据库, 具体实现逻辑如下。

第一步: 申请数据对象:

```
private static MySqlConnection mySqlConnection;//连接类对象  
private static string host;//IP地址  
private static string port;//端口号  
private static string userName;//用户名  
private static string password;//密码  
private static string databaseName;//数据库名称
```

26

第二步: 创建连接类对象并在构造函数中赋值, 代码如下:

```
public MySqlAccess(string _host, string _port, string _userName, string  
_password, string _databaseName) {  
    host = _host;
```

```
port = _port;
userName = _userName;
password = _password;
databaseName = _databaseName;
OpenSql();//打开数据库
}
```

通过这个构造方法即可指定数据库的访问对象，在验证数据库的用户名，密码以及访问的数据库存在后会去执行OpenSql函数，该函数是连接数据库的具体执行者，实现代码如下：

```
public void OpenSql() {
    try {
        string mySqlString =
            "Database=" + databaseName + ";DataSource=" + host + ";User=" + userName + ";Password=" +
password + ";port=" + port + ";";
        MySqlConnection = new MySqlConnection(mySqlString);
        UnityEngine.Debug.Log("连接成功");
        MySqlConnection.Open();
    }
}
```

OpenSql方法的以上代码片段是去尝试按照用户构造的数据库名，数据库登录用户名，密码以及端口号去连接数据库，若验证成功则显示连接成功，并且打开数据库连接。此时系统就已经可以从数据库读取数据表的数据了，若连接失败则去处理连接异常的逻辑，代码如下：

```
catch (Exception e) {
    throw new Exception("数据库连接失败，请重新检查MySQL服务是否打开。" + e.Message.ToString());
}
}
```

上述代码片段会在数据库连接异常后抛出一个异常，异常信息为数据库连接失败。

27

数据查询

数据查询时会去调用一个名为Select的方法，该方法包含的参数为数据表名，查询数据项，查询的数据的列名，操作符以及查询值，在执行该方法时，首先进行安全校验，检测查询语句是否符合规范，方法具体如下：

```
public DataSet Select(string tableName, string[] items, string[] whereColumnName, string[] operation,
string[] value) {
    if (whereColumnName.Length != operation.Length || operation.Length != value.Length) {
        throw new Exception("输入不正确：" + "要查询的条件、条件操作符、条件值的数量不一致！");
    }
}
```

方法的以上代码片段进行语法规则的检测，若不符合查询语句的规范则会抛出异常信息。之后进行数据的查询逻辑，如下所示：

```
string query = "Select " + items[0];
for (int i = 1; i < items.Length; i++) {
    query += "," + items[i];
}
```

```
query += " FROM " + tableName + " WHERE " + whereColumnName[0] + " " + operation[0] + " '" +  
value[0] + "'";  
for (int i = 1; i < whereColumnName.Length; i++) {  
    query += " and " + whereColumnName[i] + " " + operation[i] + " '" + value[i] + "'";  
}
```

上述代码片段会将作为参数产地进来的查询信息拼装成一个查询语句，拼装完成后会去调用下面的QuerySet方法，该方法的参数为查询语句和语句类型，下面的示例为Select类型，该方法返回DataSet类型的数据结果。

```
return QuerySet(query, Operate.Select);  
}
```

数据插入

数据插入方法会去执行Insert方法，该方法的参数包含插入的数据表名称，

28

插入的数据项以及插入的值，之后去判断需要插入的数据在数据表中是否存在，若存在则无法插入，返回false值，若不存在则去执行数据插入操作，Insert

方法，具体代码如下：

```
public bool Insert(string tableName, string[] items, string[] value) {  
    DataSet ds = QuerySet("Select user_id from users where user_name='" + value[0] + "'",  
Operate.Select);  
    if (ds != null) {  
        return false;  
    }  
}
```

方法的以上代码片段是检测数据项是否在数据库中存在的逻辑，若可以插入则去执行下面的插入逻辑。

```
else {  
    if (items.Length != value.Length) {  
        throw new Exception("输入不正确： " + "要查询的条件、条件操作符、条件值的数量不一致！");  
    }  
    string query = "Insert Into " + tableName + " ( " + items[0];  
    for (int i = 1; i < items.Length; i++) {  
        query += "," + items[i];  
    }  
    query += " ) Values ( '" + value[0] + "'";  
    for (int i = 1; i < value.Length; i++) {  
        query += "','" + value[i] + "'";  
    }  
    query += " )";  
    UnityEngine.Debug.Log(query);  
    QuerySet(query, Operate.Insert);  
}  
return true;
```


上述代码片段在检测了插入语句符合规范后，将参数传递进来的插入信息拼装成数据插入语句，之后去执行拼装好的插入语句，执行者为QuerySet(query, Operate.Insert)方法。

```
}
```

执行数据库语句

29

上述数据查询和插入方法都提到了一个名为QuerySet的方法，该方法是sql语句的实际执行者，该方法参数为sql语句和语句的类型，具体实现代码如下：

```
private DataSet QuerySet(string sqlString, Operate op) {
```

先检测数据库连接是否为打开状态，若为打开状态则申请一个DataSet类型的对象ds用来存储sql语句执行结果。

```
    if (mySqlConnection.State == ConnectionState.Open) {  
        DataSet ds = new DataSet();  
        MySqlDataAdapter mySqlAdapter = new MySqlDataAdapter(sqlString, mySqlConnection);  
        mySqlAdapter.Fill(ds);
```

上述代码片段申请了一个MySqlDataAdapter 类型的对象，该对象的功能是负责向ds中写入sql语句执行结果。之后根据需要执行的sql语句类型去执行具体对应的操作，下方示例为select语句和insert语句，执行之后将ds对象返回，并关闭写入连接。

```
        switch (op) {  
            case Operate.Select: return ds.Tables[0].Rows.Count > 0 ? ds : null;  
            case Operate.Insert: return null;  
        }  
        mySqlAdapter.Dispose();  
    }  
    return null;  
}
```

数据库关闭

在执行完数据库的操作后，为了减少系统的资源占用率，需要关闭不需要的数据库连接，关闭操作主要是由CloseSql方法来执行，具体的操作如下：

首先通过检测数据库连接对象是否存在来进行安全校验，若存在表示可以关闭，通过mySqlConnection != null这句代码来判定，若部位空值则进行后续的关闭操作，代码如下：

```
mySqlConnection.Close();
```

```
    mySqlConnection.Dispose();
```

```
mySqlConnection = null;
```

30

关闭数据库连接，释放连接占用的资源，将数据库连接对象置空，通过这三个步骤即可将数据库连接占用的资源释放掉，节省系统的资源占用比。

31

用户界面

6.1 注册登录界面

登录注册界面包含两个主要功能，即用户的注册和登录，登录界面如图6.1所示：

图6.1

32

6.2 菜单界面

菜单界面包含主要是用户进行功能选择及界面跳转的功能，菜单界面如图6.2所示：

图6.2

6.3 成绩查询界面

进入成绩查询界面点击查询按钮即可查询到学生各个学期的所有成绩信息，

33

成绩查询界面如图6.3所示:

图6.3

6.4 课程查询界面

进入课程查询界面点击查询按钮即可查询到学生本周的课程信息表,学生可以在课程信息界面查看自己最近一周的课程安排,课程查询界面如图6.4所示:

34

图6.4

6.5 校园资讯查看界面

进入校园资讯界面查看最近学校的动态和新闻,学生可以在这里看到资讯的标题图片和内容概述,然后选择自己感兴趣的资讯点击查阅,校园资讯界面如图6.5所示:

35

图6.5

6.6 调查问卷界面

进入调查问卷查看学校发布的调查问卷,可以填写完成后点击提交按钮完成提交,调查问卷界面如图6.6所示:

36

图6.6

总结及反思

本系统是一款基于Android的掌上校园应用，本文的叙述包含了该系统的基础功能和模块，实现了学生成绩查询，课程查询，校园资讯查阅，调查问卷提交这四个主要功能，在日常的学习生活中提高了学生的学习效率，拓宽了获取信息的途径，本系统界面简洁，操作较为简单，利用该系统可以方便的查询学生个人信息，虽然功能不算太多，但是实用性还是较为可观的，有利于信息化校园的建设。

在设计本系统时，先分析系统的业务流程，分析出该系统的需求和业务逻辑，接着分析系统的功能，对系统功能进行建模，分析系统所需的数据，设计数据库对系统数据做以归纳，其次进行概要设计，主要包含系统的接口，结构以及处理错误的逻辑等，接下来进行详细设计，具体到每一个功能模块的设计，最后进行编码和测试，相对于所浏览的一些文献资料，我规划的路线中将设计数据库这一步骤没有放在详细设计的环节，而是在分析系统数据时进行设计，我认为这样有利于数据库设计的准确性和合理性，使其可以有效快速的实现数据查询的功能，然后在使用Unity开发工具的基础上，搭建了GUI界面，通过引入数据库和json链接库的方法，实现了读取数据库数据和存取文件的功能，最后使用UDP协议的Socket实现了客户端和服务端的通信，完成了用户的功能需求，本系统结合了实际情况，从用户的角度出发，考虑用户的实际需求和使用体验，针对西北大学的具体情况和学生用户的需求，它很好的融合了学校咨询周边的信息，又可以查询到自己的相关课程信息，对于老师或者活动发起人而言，它的调查问卷功能又可以很好的帮助用户了解详细的情况，便于之后的工作和活动的开展进行。最后使之成为一个可以便于学生和老师使用，准确的实现需求和功能的APP。

通过这次的软件设计和实现，我学到了很多知识，包括对C/S架构也有了新的认知，在后续的分析和测试时，我也发现了系统的一些不足和我在设计阶段存

在的问题和缺陷，通过分析这些问题我认识到了系统设计需要考虑的因素并不是想象中的那么简单，每一个模块之间的耦合性和功能之间的关联都需要慎重的去构思和实现，这对系统设计人员的逻辑思维有着很高的考验能力，由于个人能力的一些原因，本系统在一些方面没有达到预期的效果，比如数据的存储使用json格式很容易泄露出去，数据传输过程中的数据也都是以明文的形式传输，在安全性上做的还不够好，本次系统的设计实现也让我看到了自己在软件设计方面的缺点和不足，上述提到的这些问题都是以后开发生活中我需要去考虑和改进的地方，我相信只要认真考虑和学习，在之后的开发生活我会改正这些问题的。

参考文献

- 芦雪莲, 李勇. 校园APP应用对高校学生管理工作的影响性研究[J]. 思想政治教育研究, 2016, 32(05):113-115.
- [1] 郑磊. 基于三维网页技术的Unity3D教学管理系统的设计与实现[D]. 上海交通大学, 2013.
- [3] 黄晟. 基于用户体验的APP设计研究[D]. 陕西科技大学, 2012.
- [4] 杨哲. 基于Android的智慧校园APP设计[D]. 哈尔滨工业大学, 2018.
- [5] 闫素旺. 基于安卓平台的掌上校园系统的设计与实现[D]. 河北工业大学, 2013.
- [6] 于珊. Android App快速开发平台设计与实现[D]. 西安电子科技大学, 2015.
- [7] 苟文博, 于强. 基于MySQL的数据管理系统设计与实现[J]. 电子设计工程, 2017, 25(06):62-65.
- [8] 任海鹏, 邓春红, 方继才. 智慧校园移动系统数据库设计[J]. 安阳师范学院学报, 2018(02):46-48.
- [9] 胡敏. Web系统下提高MySQL数据库安全性的研究与实现[D]. 北京邮电大学, 2015.
- [10] 曾露. MVP模式在Android中的应用研究[J]. 软件, 2016, 37(06):75-78.
- [11] 邹建国, 马丁, 刘奇祥. 基于B/S与C/S架构的高校社团管理系统[J]. 现代计算机, 2013(19):58-62.
- [12] 黄永华. 应用Socket的微服务之间的通讯[J]. 福建电脑, 2020, 36(02):68-71.
- [13] 杨辉, 王建新. 基于C/S和B/S结构的高校教务管理系统的设计与实现[J]. 湖南工业大学学报, 2009, 23(06):69-72.

说明:

1. 送检文献总字数=送检文献的总字符数, 包含汉字、非中文字符、标点符号、阿拉伯数字 (不计入空格)
2. 总相似比=送检论文与检测范围全部数据相似部分的字数/检测总字符数
3. 参考文献相似比=送检论文与其参考文献相似部分的字数/检测总字符数
4. 辅助排除参考文献相似比=总相似比-参考文献相似比
5. “单篇文献最大相似比”: 送检文献与某一文献的相似比高于全部其他文献
6. “是否引用”: 某一相似文献是否被送检文献列为其参考文献