**南 开 大 学**

**滨 海 学 院**

本 科 生 毕 业 论 文（设 计）

中文题目： 基于Java的教务网移动端的安卓开发的设计与实现

外文题目：Android development of mobile terminal of educational network’s design and implementation based on Java language

学 号： \_ 15990046\_\_\_\_\_

姓 名： \_\_\_蔡箐菁\_\_\_\_\_\_

年 级： \_\_\_2015级\_\_\_\_

专 业： 数字媒体技术

系 别： 计算机科学与技术系

指导教师：\_\_\_\_\_\_ 刘嘉欣\_\_\_\_

完成日期： 2017年12月29日

关于南开大学滨海学院本科生毕业论文（设计）的声明

本人郑重声明：所呈交的学位论文，是本人在指导教师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或没有公开发表的作品内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：

年 月 日

本人声明：该学位论文是本人指导学生完成的研究成果，已经审阅过论文的全部内容，并能够保证题目、关键词、摘要部分中英文内容的一致性和准确性。

学位论文指导教师签名：

年 月 日

摘 要

本程序完成了从登录教务网（记住密码的功能），修改教务网密码，从教务网获取成绩信息，修改个人头像，从自己开发的服务器端拉取使用者的个人信息和查看通讯录与在数据库里存过信息的人实现聊天的功能。另外，自己开发的服务器端连接的数据库里的信息和教务网信息是不相匹配的，全部都是我制造的假数据。

服务器端使用的开发工具有MyEclipse2014、Hibernate、Tomcat,服务器端的脚本语言使用的是JSP，连接的数据库是MySQL.

客户端的使用的开发工具是Eclipse，实现UI界面的设计，和用户交互功能。

关键词： 教务网；移动端版；JSP开发的服务器端；MySql；聊天功能；

**Abstract**

Completed the program from the login teaching network with persistent password, from educational network performance information, from their own development of server side pull the user's personal information and the information saved in the database of people can realize the function of chat. People also can modify themselves passwords in educational networks and avatars. In addition, their development of the database server connection information and educational network information is not matching, all is I make false data.

Server-side development tools used are MyEclipse2014, Hibernate, Tomcat, server-side scripting language is used JSP, connect the database is MySQL.

The client's use of development tools is the Eclipse, UI design, and user interaction function.

**Keywords：**educational network; Mobile version; Server side of JSP development; MySql; Chat function;

**目录**

[第一章 绪论 1](#_Toc502697287)

[第一节 引言 1](#_Toc502697288)

[第二节 课题描述 2](#_Toc502697289)

[第三节 国内外的研究 4](#_Toc502697290)

[第四节 论文组织结构 4](#_Toc502697291)

[第二章 关键技术及理论基础 5](#_Toc502697292)

[第一节 开发使用的关键技术 5](#_Toc502697293)

[一.Cookies的存储 5](#_Toc502697294)

[二．异步任务的使用 6](#_Toc502697295)

[三.Hibernate 7](#_Toc502697296)

[第二节 实现登陆教务系统和实时聊天的理论基础 8](#_Toc502697297)

[一．HTTP协议简介 8](#_Toc502697298)

[二．发送请求 8](#_Toc502697299)

[三.接收响应 9](#_Toc502697300)

[第三章 总体设计及架构分析 9](#_Toc502697301)

[第四章 系统详细设计 10](#_Toc502697302)

[第一节 服务器模块 10](#_Toc502697303)

[第二节 客户端模块 12](#_Toc502697304)

[第五章 系统实现及测试 13](#_Toc502697305)

[第一节 服务器端实现 13](#_Toc502697306)

[一．查询信息 13](#_Toc502697307)

[二．更新信息 14](#_Toc502697308)

[三．增添信息 14](#_Toc502697309)

[四．查询最新的聊天记录 15](#_Toc502697310)

[第二节 客户端控制应用实现 16](#_Toc502697311)

[一．向教务网所在服务器发送请求 16](#_Toc502697312)

[二．向自己开发的服务器发送请求 18](#_Toc502697313)

[三．对ListView的优化之一：converView 19](#_Toc502697314)

[四．对ListView的优化之二：ViewHolder 20](#_Toc502697315)

[第六章 总结与展望 21](#_Toc502697316)

[致谢 23](#_Toc502697317)

1. **绪论**

## 第一节 引言

由于我们学校一直都只有web端的教务网，我在使用的时候总会觉得有诸多

不适。比如一页上的内容太过复杂，但是我在实际使用上有很多功能我根本就使用不到。另外由于手机屏幕的限制，出现在浏览器上打开的网页字太小，操作起来不方便的情况。在手机上浏览教务网网页还有一个问题就是会出现，就是想要回退的时候就非常不方便了。除非每一个页面上都有相互跳转的超链接，但是这样的界面看上去很乱，这又导致了第一个问题的出现。

就目前来说，由于我们的社交环境、生活环境都相对局限，而像微信这样的线上聊天功能已经取代了短信成为了我们主要的保持社交和获取和交换信息的方式。但是在浏览器和微信之间相互切换也是一件相当恼人的事情。

出于以上三个问题的考虑我设计了这个移动端版的教务网。简单地实现了聊天功能和我们常用到的查询成绩的功能。由于修改密码虽然不常用，但是我也作为一个必要的功能模块加入到软件当中，感觉这样会更加完整一些。同样地，查询个人信息（也就是“我”这一功能模块）和修改头像的功能都是出于同类软件（这里特指和微信比较）都具有的考虑上加入的功能模块，为了本款软件的完整性。

最后想做一个移动端版的教务网其实是因为开学时老师在屏幕上演示了金黎明学长的作品，很羡慕，很想试试学长能做的我能不能做。

第二节 课题描述

本软件从首页进入到登入界面再进入到选择界面，选择界面会出现四个功能模块，分别是查看个人信息（指自己开发的数据库中）、修改个人在教务网上的密码、查询真实的成绩信息、和好友实现聊天的功能。其中查看个人信息的功能模块里面加上了修改头像的功能。现在进行简单的一一介绍。

要实现登录、修改密码和查询成绩都需要先了解HTTP协议。我们浏览网页使用的就是HTTP协议，Http是一种通信协议，它允许将超文本标记语言文档从Web服务器传送到客户端的浏览器。

当我们输入URL（Uniform Resource Location）后，浏览器会给服务器发送一个Request，当Web Server接到Request后会生成相应的Response，再发回给浏览器。浏览器解析Response（Body）中的HTML文档，就变成了我们看见的网页了。

需要强调补充的是，HTTP协议是无状态的，所以同一个客户端的的这次请求和上次请求是没有对应关系的，也就是说，Web Server是不知道两次请求都来自同一个Client，因此Web程序引入了Session,Cookie来维护状态。当浏览器第一次访问某些服务器的时候，服务器会发现接收到的数据包没有携带SessionID,就会自动创建Session对象，并将创建一个保存有SessionID的cookie发送给浏览器。在这里SessionID其实就是Session对象的索引。浏览器下次再访问服务器时就会发送携带着这个SessionID的Cookie对象。

这里我需要做的工作就是模仿浏览器的行为向Web Server发送请求，然后解析从Web Server收到的Response 中的Html文档，得到我需要的信息，再填充到我的android移动端的UI界面上。

发送请求是我需要先知道浏览器向服务器发送的请求里面要包含哪些参数（一堆的key-value值），这就需要在浏览器发出请求时进行抓包，这里我使用的工具是WireShark，其实使用Chorme的F12开发者工具一样可以实现抓包。找到自己需要发送哪些参数。方便在后续的请求里面进行填写。

为了方便我使用了在Android Studio中已经实现内嵌，但是在Eclipse中还需要导入包才能使用的Okhttp,来封装请求。解析从浏览器发送回来的HTML文档我使用了JSoup。

另外由于教务网是需要登录的，所以在登陆之后的所有操作都是需要Cookie来维持状态的。Cookie时客户端技术，程序把每个用户的数据以Cookie的形式写给用户各自的浏览器。当用户使用浏览器再去访问服务器中的Web资源时，就会带着各自的Cookie去，这里需要实现Cookie的存储和发送，Okhttp都提供了相应的函数。

查看个人信息、修改头像、实时聊天、其实都是对数据库的增删改查的操作。在MyEclipse 下的Web Project下，连接MySql开发的数据库，用Hibernate反向引擎生成和数据库表相对应的Java类，通过DAO类对象中的方法可以用Java语言直接对数据库进行增删改查。其结果都以字符串的形式显示在JSP脚本编写的页面上。Android端通过URL类，以及URL类的对象的openStream()方法，可以读取到网页上的信息，然后再通过Gson转化成对应的对象。这时候就可以实现对UI控件内容的填充。

举例来说，在查看个人信息时，通过GET提交的参数取服务器端的数据库中进行数据查询。GET提交的数据放在URL之后，以‘?’分割，URL传输的参数间以‘&’相连。服务器端就是通过提取URL里的参数，作为WHERE语句里面的查询条件，进相连接的数据库中进行查询符合条件的列。同样地，修改头像就是对数据库进行一个先查询再修改的过程。

## 第三节 国内外的研究

国外对这方面的研究对这方面研究的资料比较少，因此不在我考虑的范围之内。国内网站上可以搜寻到的相关资料大多是关于方正教务网移动端登录功能的实现。有一定的参考价值。

国内有一款APP叫做“超级课表”，可以获取到每个学校的课表信息，登录是也是和登录教务网差不多，需要输入学号和密码。同时可以在该平台上发布信息，加好友相互聊天。我写的这个程序与这款软件十分相似，“超级课表”获取每个学生的课程表应该也是找到学校对应教务网发送请求，实现聊天和发布信息是在自己开发的服务器端的数据库进行增删改的操作。

## 第四节 论文组织结构

本论文主要从以下方面安排及组织文章的结构：

第一章绪论部分，首先对移动端版教务系统这一课题背景的行业背景及学术背景进行了基本的描述，在此基础上，对本课题系统的基本功能进行完整的表述。最后通过对当前国内外的应用研究场景及开发的意义上进行介绍。

第二章系统研发相关理论及关键技术介绍部分，此部分介绍了移动端版的教务系统所使用到的各种理论与关键技术。

第三章总体设计及架构部分，此部分通过对功能的详细分析与判断，把一套完整的系统进行划分，确定所要实现的多层环节和各环节的详细功能。

第四章系统详细设计部分，此部分通过程序各部分的功能模块的功能介绍和详细说明以及开发工具的介绍还有实现这些功能需要的基础知识储备。从而实现移动端版的教务系统

第五章系统实现及测试部分，此部分通过之前考察及分析登录、修改密码、修改头像、查看个人信息、查看成绩、实时聊天功能方面对实现这样一个系统进行设计与实现。

第六章总结与展望部分，这部分内容总结了本次进行的android设计完成过程所学到的东西以及所获得的心得体会。

第二章 关键技术及理论基础

## 第一节 开发使用的关键技术

### 一.Cookies的存储

Okhttp提供了相应的CookieJar这个类来实现Cookies的自动存储与发

送。当服务器端给客户端发送Cookie对象时调用saveFromResponse()，将Cookies存进我们自己创建的List对象当中。客户端向服务器发送请求时，会通过loadForRequest()的方法获取保存在list里面的Cookies，封装进请求头里发送给服务器端。

由于在实现登录以后再访问服务器的其它页面时就没有必要重新获取Cookies了，所以我在第一次获取了Cookies以后就对其进行了持久化存储。

在开发本系统一开始我就在Cookies上遇到了问题，查找bug好多天，所以在这里将Cookie放在第一。

### 二．异步任务的使用

我所知道的在Android里想要使用异步有两种方法：一种是Handler，一种是Asynctask。在Android里，主线程也叫做UI线程，在其中主要实现的是对UI控件的一些操作。在Android开发中我们必须遵守这个单线程模型的规则。另外由于Android中的UI操作并不是线程安全的，并且这些操作都要在UI线程中执行。对服务器发送请求的功能需要在网络线程中进行。如果在UI线程中进行和网络相关的操作，网络操作属于耗时操作，UI线程超过5秒没有响应用于请求，就会引发Application Not Responding Exception。Android4.0之后更是禁止了在UI线程中执行网络操作。因此在Android中实现异步操作就显得尤为重要。

当然new Thread()重新去开辟线程也是可以的，但是由于我们无法完美保证多个线程之间的执行顺序，所以就更需要异步任务来支持。

在使用Handler和Asynctask的过程中，我认为Handler在实现上是最为简单的，也是最容易理解的。就是让子线程给子线程发送消息，从子线程跳回到主线程，以便于进行UI操作。但是当进行多次连接网络的操作时，大量用Handler写出来的程序结构就会变得十分混乱了。因为要去找对应的Handler下的HandleMessage，使程序的可读性降低。

而Asynctask类很好地把UI线程和子线程都封装在一起，写出来的程序结构清晰，并且在实现进度条上与Handler相比有着卓越的方便性。在doInBackground()实现较为耗时的网络操作。doInBackgroung()执行完以后执行onPostExecute()，所有的UI操作都可以放在这个函数里面执行。另外还有onPreExecute()是在doInBackground()之前被调用的，通常用于一些初始化操作，如：进度条的显示。publishProgress()和onProgressUpdate()配合使用实现进度条的更新。

### 三.Hibernate

hibernate是一个开源框架，它是对象关联关系映射的框架，它对JDBC做了轻量级的封装，而我们java程序员可以使用面向对象的思想来操纵数据库。  
hibernate核心接口；session：负责被持久化对象CRUD操作；sessionFactory:

负责初始化hibernate，创建session对象；configuration:负责配置并启动；hibernate，创建SessionFactory；Transaction:负责事物相关的操作；Query和Criteria接口：负责执行各种数据库查询

用myEclipse反向引擎工具来由数据库里的表生成实体类和hibernate文件。我们就可以通过DAO类中提供的函数对相连接的数据库进行增删改查的操作了。

第二节 实现登陆教务系统和实时聊天的理论基础

## 一．HTTP协议简介

HTTP是一个属于应用层的面向对象的协议。HTTP允许传输任意类型的数据对象。是一个基于请求与响应模式的、无状态的、应用层的协议。

支持C/S模式，当客户向服务器请求服务时，只需传送请求方法和路径。请求方法常用的有GET、POST。每种方法都规定了客户和服务器联系的类型。另外还具有无连接、无状态的特点。其中无连接是值：限制每次连接只处理一个请求。服务器处理完客户的请求，并受到客户的应答后，即断开连接，采用这种方式可以节省传输时间。无状态是指：HTTP协议是补状态协议，无状态是指协议对于事务处理没有记忆能力，缺少状态意味着如果后续处理需要前面的信息，则它必须要进行重传，这样可能导致每次连接传送的数据量增大。

## 二．发送请求

HTTP URL包含了用于查找某个资源的足够的信息。

HTTP请求方法有多种。我只用到了GET和POST两种。其中GET是请求获取Request-URI所标识的资源。这里我用在了获取教务网默认首页和获取成绩以及获取修改密码的页面上。POST要求被请求的服务器接受附在请求后面的数据，常用于提交表单。

## 三.接收响应

在接收和解释请求消息后，服务器会返回一个HTTP响应消息。 HTTP响应有三个部分组成：状态行、消息报头、响应正文。其中Cookies在消息报头，而传回来的HTML文档在响应正文里。

接收到响应后浏览器再进行解析，就生成我们看到的网页页面了。

第三章 总体设计及架构分析

按照模式来划分可以将本系统分为两个部分。

第一个部分是通过发送网络请求向教务网所在服务器发送请求，然后解析发回来的HTML文档。获取表格里面的数据。

另一个部分就是实现实时聊天的功能，这个功能必须要自己开发服务器端，自己开发数据库，每发布一条聊天消息都相当于是往数据库里面存储了一条信息。聊天界面的刷新，就相当于从服务器里面不断地读取消息。

本程序采用的是C/S架构的三层结构。客户端接收用户的请求，客户端向应用服务提出请求，应用服务从数据库服务中获得数据，应用服务将数据进行计算再将结果提交给客户端，客户端将结果呈现给用户。再三层结构当中，客户端并不参与运算，只是简单地接收用户的请求，显示最后的结果。由于三层结构中的客户端并不需要参与运算，所以对客户端的配置要求是比较低的。

第四章 系统详细设计

经过前一章系统架构分析后发现，由于本次设计的系统是一套不同设备间的多平台系统，所以系统详细设计部分依然按照模块化的功能设计来完成。本系统的功能模块划分如图4-1。

传送数据

浏览器

服务器

传回处理过的数据

发送数据

客户端

图4-1 模块划分示意图

1. 服务器模块

服务器模块是对从URL里获取到的参数进行处理。包装成HQL语句对数据库进行查询，然后回显到页面上。客户端获取到的数据又都是从服务器网页网页上直接拽取下来的字符串。本系统的主要数据处理都是在服务器端进行。服务器模块又主要包含两大功能，一个是对数据库进行增删改查的工作，另一个是数据展示。

包装成HQL语句对操作数据库

获取URL里的参数

将查询结果回显到网页上

4-2 服务器处理数据及显示流程示意图

查找新存入的聊天记录

查找登录者的所有朋友

查找数据库里指定的聊天记录

按照头像编号查找头像信息

查找数据库里的所有头像资源

查找数据库里特定使用者信息

发布聊天消息

更新数据库中的使用者信息

服务器功能

图4-3 服务器功能模块示意图

## 第二节 客户端模块

客户端需要完成的功能就是按照使用者的需求向服务器发送请求，然后服务器返回筛选好或者反应操作是否成功的数据，客户端接收到以后对这数据进行解析，再显示到UI控件上，显示给使用者。

收到服务器返回的数据

显示到UI组件上

向服务器提交GET/POST请求

将查询条件整合进URL中

图4-4 客户端版发送数据接收数据流程示意图

客户端功能模块

登陆页面

查看成绩信息

查看个人信息

查看通讯录信息

修改个人头像

进行实时聊天

修改密码

图4-5 客户端功能模块示意图

# 第五章 系统实现及测试

## 第一节 服务器端实现

### 一．查询信息

在MyEclipse端连接数据库，通过MyEclipse的Hibernate反向引擎生成对应数据库表的实体类。一共有两种类型的类生成，一种是“表名.java”是与数据库表格的一一对应。另一种是“表名+DAO.java”数据访问对象类，是一个面向对象的数据库接口，通过这个类生成的对象可以间接地对数据库进行增删改查的工作。

我另外按照每个实体类又写了相对应的服务类，旨在能将实现一种我需要的功能的语句封装在一起。为在写JSP脚本的时候调用方便。

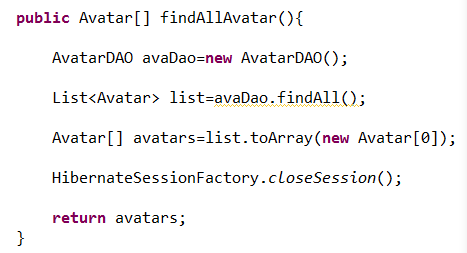


图5-1 查找所有头像功能示意代码图

以下是根据具体条件查询数据的功能代码部分：

Query query=chatlogDao.getSession() .createQuery("from Chatlog

where ((senderID=:uid1 and receiverID=:uid2)

or (senderID=:uid2 andreceiverID=:uid1)));

query.setString("uid1", senderID);

query.setString("uid2", receiverID);

其中createQuery()当中的字符串是HQL语句。但是最后会被转换成SQL语句对数据库进行操作。另外这段代码使用了的是动态绑定参数的方法，query.setString(“uid1”,senderID)就是将senderID的值赋给uid1。

### 二．更新信息

类似于要实现修改密码或是修改头像一类的功能时，就是对数据库内的信息进行更新。由于本程序中咋修改密码这一块直接是通过发请求要求学校教务网所在服务器进行修改，所以不涉及到数据库的操作。但是头像信息是存在我自己开发的数据库中的，这就需要涉及到更新操作了。

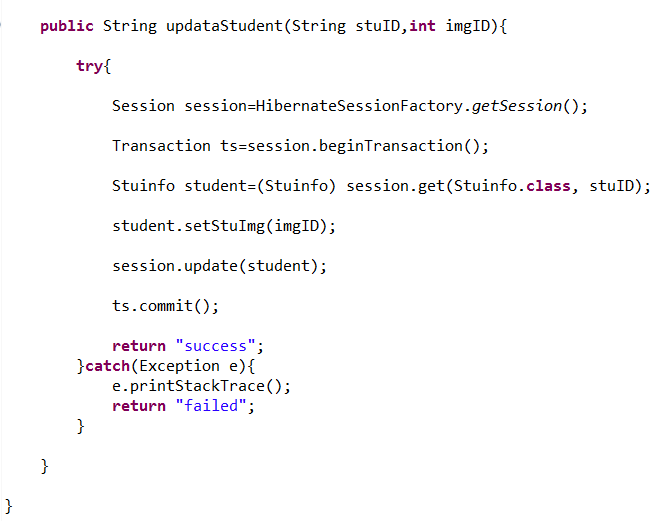


图5-2 更新头像信息示意代码图

### 三．增添信息

在本系统中要完成发布聊天信息这个功能模块时就需要用到往数据库里增添信息的功能。由于已经在登录模块完成过登录，所以可以通过Bundle可以直接将学号信息传入各个Fragment当中。聊天对象ID的获取则是通过点击list item，得到对应聊天者的姓名，再取数据库中查找出对应的ID。聊天内容从EditText当中获取，就可以组成一个URL向服务器发送请求了。

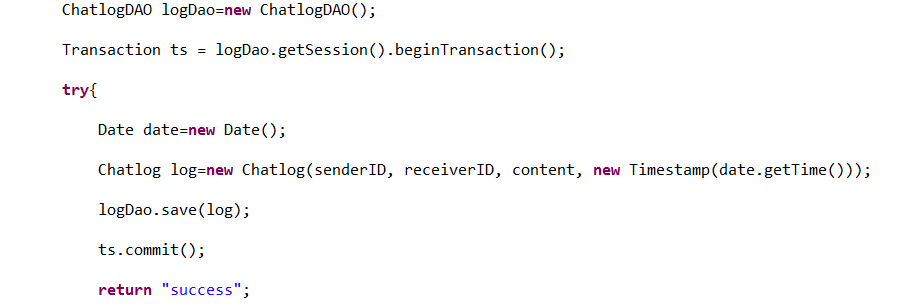


图5-3 发布聊天信息示意代码图

### 四．查询最新的聊天记录

这个功能模块放在最后是因为为提高程序性能而做出的优化。前三项只是概括代表了我在本程序中用到了哪些数据库操作。

这里的“最新”指的是当前登录下，最近一次获取过的聊天记录中的最后一条的下一条记录。微信里实现聊天时，能在界面灵活地收取到来自对方发来的消息。应该是一直在对数据库进行读取。这里我改成了当点击一次发送按钮的时候对数据库的信息进行一次读取。

在这里刷新页面有两种实现方法，由于聊天界面是通过ListView来实现的，所以每一条聊天记录都是一个item.这些item中的数据又是来自一个List。第一种方法就是每次都清空列表然后，然后按照senderID和receiverID进行查询。再将查询结果全部塞进List。通过setAdapter()显示到聊天界面上。但是这样做的缺点显而易见了，随着聊天次数的增多，每次要拉取的信息也会越来越长，所以这种方法不可取。第二种方法就是拉取最近的聊天信息。也就是上次啦去过的聊天信息的最后一条的下一条信息。实现代码如下。



图5-4 查找最新聊天信息示意代码图

第二节 客户端控制应用实现

### 一．向教务网所在服务器发送请求

当我们需要登入系统时，需要先获取到教务网的系统登录页面，这时需要用到GET方法取请求页面。Okhttp已经提供了响应的接口函数。这里我将OkHttpClient对象封装封装进storeInfo当中，方便其它Activity中需要进行网络操作的类进行请求时调用。

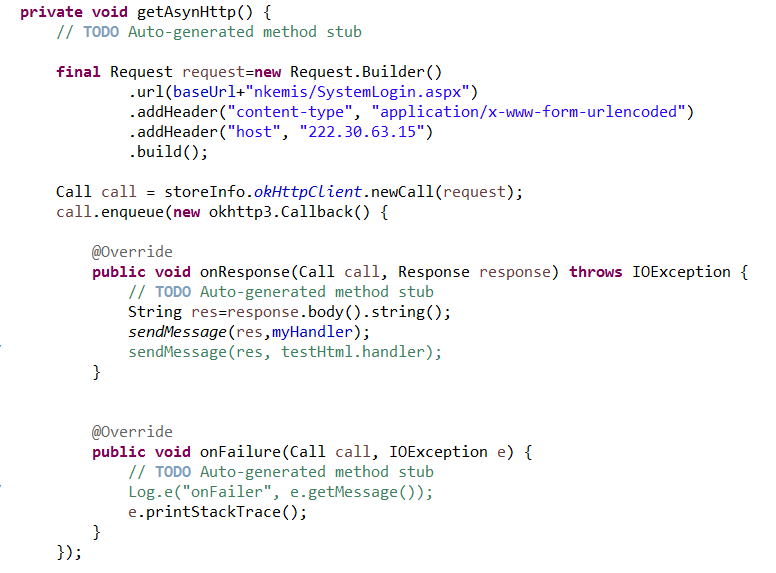


图5-5 异步GET请求默认页面

完成对默认页面的请求之后，从Response中的body中提取，我先在浏览器上完成登录操作，然后用WireShark进行抓包，查看POST方法提交的表单参数。然后再用JSoup去默认页面的表单中提取相应的参数，即图5-6中的\_\_EVENTVALIDATION、\_\_VIEWSTATE、ImageButton1.x、ImageButton1.y、txtPasswd、txtUserID,中的这六个参数，是需要我们封装进请求体里面的。成绩查询时则只需要用异步GET的方法了。

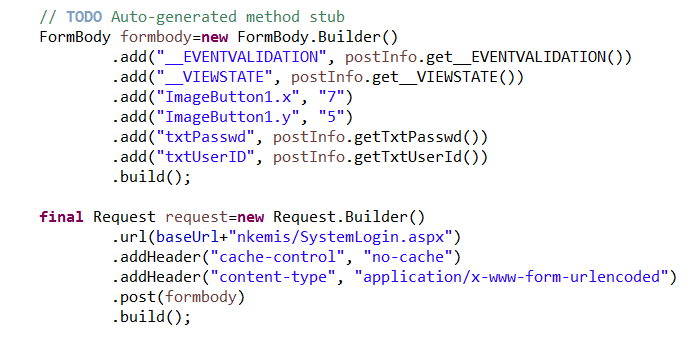
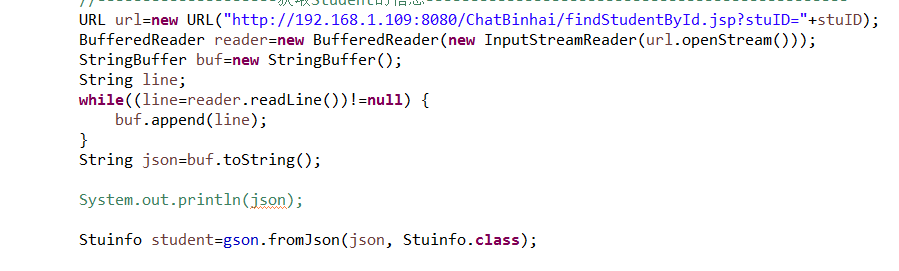


图5-6 异步POST提交

### 二．向自己开发的服务器发送请求

当使用者需要使用更改头像和查询个人信息以及实现实时聊天的时候都要向我们自己开发的服务器进行默认GET请求。图5-7所示的是查询个人信息时所用到的核心代码。



5-7 使用URL直接进行

从图5-6可以看出，GET请求也是可以携带参数的，参数直接写在URL当中，参数与参数之间以&进行连接。在JSP写的脚本中会有一个函数进行提取URL中的参数值：request.getParameter("stuID")。

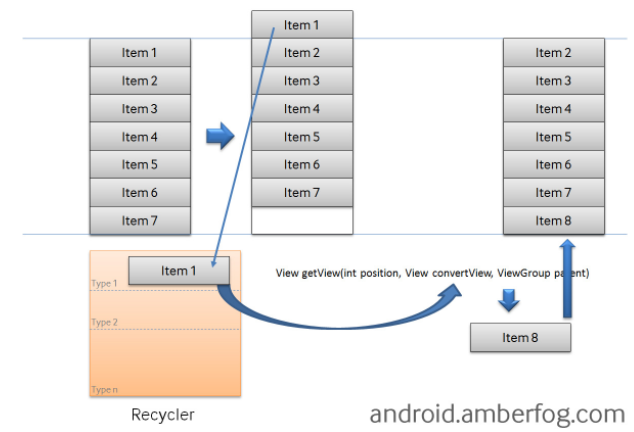
但是GET请求的参数值都是暴露在URL里面的，POST请求提交的参数都是封装在请求体里面的，不易被人获取。因此POST提交参数比GET要更加地安全。

修改头像和实时聊天的网路请求部分都和上述使用方法差不多，只不过是URL的不同，调用不同的资源。因为对数据库查增删改查的工作，都由服务器替我们完成了。

### 三．对ListView的优化之一：converView

ListView这个UI组件在这个程序里被使用的次数非常多。和异步任务一样很必要写出来。ListView针对List中的每一个item，要求adapter个“给我一个视图（getView）”，然后一个新的视图被返回显示。但是如果我们有很多个item怎么办？一一为每个item创建一个新视图显然是不可能的。

Android中有一个叫做Recycler的构建。图5-8是Recycler的工作原理。



5-8 Recycler的使用原理

Android的ListView在初始化的时候，加入整个手机屏幕最多只能放下7个item，就只会创建7个视图。创建好的视图就是getView()中的convertView参数。如果我们继续向上滑动，item1消失，item8会出现，系统不会为item8再创建新视图，item1已经划出手机屏幕了，item1的convertView正好拿出来给item8使用。总结来说，converView的功能就是将之前加载好的布局进行缓存，方便之后可以进行重用。





图5-9 converView使用代码图

在getView()中进行了判断，如果convertView为空，则使用LayoutInflater去加载布局，如果不为空则直接对converView进行重用。这样就大大提高了ListView的运行效率。可以在快速滚动的时候表现出更好的性能。

### 四．对ListView的优化之二：ViewHolder

目前代码已经不会再去重复地加载布局了，但是每次再getView()还是会调用View的findViewById()获取一次控件的实例。于是借助一个ViewHolder来对这部分的性能进行优化。

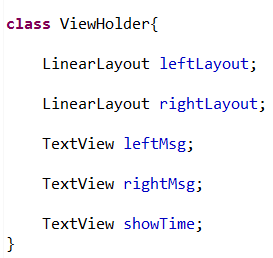


图5-10 ViewHolder类图

这个ViewHolder类的对象对控件的实例进行了缓存，当convertView为空时，创建一个ViewHolder对象，并将控件的实例都存放在ViewHolder里，然后调用View的setTag()方法，把ViewHolder对象存储在View当中。当convertView不为空的时候则调用View的getTag()方法，将ViewHolder对象重新取出来。这样所有的控件实例都缓存好了。就没有必要每次都通过findViewById()来获取控件实例了。

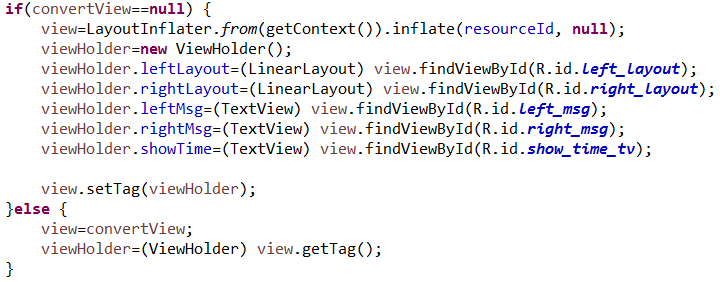


图5-11 ViewHolder使用核心代码图

第六章 总结与展望

这样一个系统做下来，感触颇多。因为遇到的错误实在太多了。清晰地记得在向自己开发的服务器发送请求时，怎么都连不上的痛苦。由于网络相关知识的欠缺，思考问题十分局限。幸好在老师创建的答疑群里有一位同学为我打开了思路。原来是连接服务器的URL中的主机名写错了。连的是无线局域网，就应该是用无线局域网下的动态IPV4,加入是插的网线，就应该写以太网下的动态IPV4。额外地，学习到了ping命令，顺带学会了怎么用笔记本发射无线信号。由于这个错误查找了许多博文，了解到多种可能发生这个错误的可能原因。可以说这一个错误带给我的收获颇丰。

这次是我第一次完整地做出一个连数据库和服务器的系统。但是一路上遇到的问题越多，到最后反而愈能克服心里对复杂的代码结构的恐惧。我对从服务器的开发到客户端的开发已经已经有一个简单的认识。美中不足的是，Hibernate的的映射方式我还没有掌握，甚至可以说完全不懂。使用MyEclipse的Hibernate反向引擎固然十分方便快捷不易出错，但是由于不懂原理，我无法在数据库设计阶段对其完整性做出设计，因为反响引擎生成出来的实体类我看不懂。由映射文件带出来的还有注解。我觉得也有必要对其进行学习和理解。

OkHttp的使用让我查阅了大量Http协议相关的资料。实现侧边栏效果找到的工具DrawerLayout的同时也让我附带知道了很多非常有创意的侧边栏设计。

程序还有许许多多需要完善的地方，特别是我一开始完成异步时使用的是Handler，后来想尝试使用Asynctask，发现用Asynctask写出来的代码结构更加清晰而且不会很乱糟糟的。我没有修改Handler部分的代码。

还有就是Activity和Activity之间，Activity和Fragment之间的参数传递，全局变量的滥用。这些都是需要再进行改进的。另外就是用户输入数据的限制，比如该输入学号的地方就不应输入英文字母。这类的安全性检查也是需要再进行改进的。

缺陷实在太多。

**参考文献**

**[1] 郭霖．第一行代码（第1版）：人民邮电出版社，2014**

**[2] eoe移动开发者社区组. Android开发入门与实战 (第二版) 人民邮电出版社,2013.**

致谢

还记得第一节课刘嘉欣老师展示金黎明学长的作品，老师的赞美之情溢于言表。觉得很羡慕，想试试学长能做的我能不能做，遂选了这个题目。老师讲课的方式，通俗易懂，追本溯源。并不是单纯地讲怎么使用，而是会讲出于什么样的问题，或是有什么样的需求导致的什么解决方法的出现，特别是会讲解一些android的内部机制，循循善诱，真的使我获益良多。

感谢这一个学期以来老师的教导，总是积极回应同学们在学习上遇到的问题，不管多简单的问题都不会感到不耐烦，还总是分享一些技术难点。虽然已经说过一次了，但还是想再说一次，真的获益匪浅。谢谢刘嘉欣老师，希望您身体健康，希望自己以后能一直保持对写码的热情。

