

“自习控”团队项目设计报告

竞赛名称：华中农业大学互联网应用技能竞赛

项目名称：基于 Android 的校园自习教室 app 开发

报告名称：“自习控”团队项目设计报告

指导老师：黄慧

专业班级：信管 1201, 1202, 1401

小组成员：肖玮：2012308200712

卢兴威：2012306202306

张良翼：2014315200530

一、团队组织结构介绍与分工

1. 团队名称：自习控

2. 团队成员：组长：肖玮（信管 1201）

组员：卢兴威（信管 1202）、张良翼（信管 1401）

3. 团队分工：

肖玮：进行软件主体设计，服务器的搭建及关键代码设计。

卢兴威：进行 app 界面设计，文档、报告的编写等。

张良翼：进行数据库数据的编写与设计，UI 美化等。

二、技术路线

项目说明：简单的 Android 应用程序通过 webservice 连接 sqlserver 数据库，具有进行简单登陆、查询功能。其中 Android 应用程序为前台，iis 上发布的访问数据库的 webservice 为后台。

Android 端：Eclipse+ADT 开发，采用 Java 编程。

服务器端：Microsoft Visual Studio 2010 开发 Webservice，采用 C#构造类。

数据库：SQL Server Management Studio

发布：iis7.0

三、实现方案流程和原理

3.1 各平台环境的搭建

3.1.1 SQL Server

开始在 MySQL 与 sqlserver 之间犹豫数据库的选择，因 MySQL 中 Doc 式建表较为繁琐，决定采用 Sql Server。Sqlserver 配置较为简单，以本机服务器作为数据库引擎，我的本机名为 jamingocci。

3.1.2 VS

由于先前信息系统开发与设计实习周采用 VS 进行设计，因此很大程度上有所熟悉。

Webservice 只需新建 ASP.NET Web 服务应用程序项目即可。

3.1.3 IIS

由于先前网站设计与开发实习周采用 IIS 进行过发布，因此很多设置与环境已经搭建。

3.1.4 Eclipse + ADT

由于步骤复杂繁多，需要安装的程序众多，且资源下载难度等问题，此为环境搭建中的重点与难点。基本步骤主要如下：

1 JDK 安装与环境变量 Path 的设置

2 Eclipse 安装

3 Android SDK 安装

4 ADT 安装

由于谷歌一些服务器国内无法下载，Android SDK 的更新与 ADT 的下载出现困难。因此最终我们还是有幸搜索到了一个集成好 ADT 的 Eclipse 下载，并且在修改了 Option 中的端口号与解析地址之后，SDK 才能够正常下载，虽然下载速度非常缓慢。在经过近一个周的环境搭

建之后，终于能够成功在模拟器中运行 HelloWorld 程序。

3.2 建造数据库

3.2.1 数据库名称：TeachingBuilding

表设计

表名称：Date01_01、Date01_02... 等（统一为 Date+当日时间格式）

表说明：

列名	中文名称	数据类型	必填	说明
Classroom	教室名	varchar(50)	V	主键
C12	12 节课	varchar(50)		
C34	34 节课	varchar(50)		
C56	56 节课	varchar(50)		
C78	78 节课	varchar(50)		
C910	910 节课	varchar(50)		
C1112	1112 节课	varchar(50)		

部分数据如下：

classroom	c12	c34	c56	c78	c910	c1112
二教 304	空	有课	空	有课	有课	有课
二教 305	空	空	有课	空	有课	有课
三教 A101	有课	有课	有课	有课	有课	空
三教 A102	有课	空	有课	有课	空	空

3.2.2 数据库名称：UserCheck

表设计

表名称：UserInfo

表说明：

列名	中文名称	数据类型	必填	说明
username	用户名	varchar(50)	V	
password	密码	varchar(50)	V	
ID	编号	Int	V	主键

部分数据如下：

username	password	ID
admin	123	1
jamingocci	1234	2

3.3 建造服务器

经过查阅大量资料，尤其是参考《android 开发与应用》（人民邮电出版社 张荣 编）后，决定使用 webservice 作为服务器工程，而没有采用 tomcat。

3.3.1 Webservice 建造步骤：

1. 新建 ASP.NET Web 服务应用程序项目。
2. 构建与 sqlserver 数据库的连接。

3. 编写*.asmx 文件，创建方法。

3.3.2 Webservice 项目名称: ClassroomManageWebServices

Service1.asmx 分别调用了教室查询的 7 个方法:

```
public class Service1 : System.Web.Services.WebService
{
    DBOperation dbOperation = new DBOperation();

    [WebMethod]
    public string HelloWorld()
    {
        String time = DateTime.Now.ToString("MM_dd");
        return time;
    }

    [WebMethod(Description = "获取所有教室的信息")]
    public string[] selectAllInfor()
    {
        if (dbOperation.selectAllInfor().ToArray() == null)
            return dbOperation.dbnnull().ToArray();
        else
            return dbOperation.selectAllInfor().ToArray();
    }

    [WebMethod(Description = "1-2 节课信息")]
    public string[] searchClassroom12()
    {
        if (dbOperation.selectAllInfor().ToArray() == null)
            return dbOperation.dbnnull().ToArray();
        else
            return dbOperation.searchClassroom12().ToArray();
    }

    [WebMethod(Description = "3-4 节课信息")]
    public string[] searchClassroom34()
    {
        if (dbOperation.searchClassroom34().ToArray() == null)
            return dbOperation.dbnnull().ToArray();
        else
            return dbOperation.searchClassroom34().ToArray();
    }

    [WebMethod(Description = "5-6 节课信息")]
    public string[] searchClassroom56()
    {
        if (dbOperation.searchClassroom56().ToArray() == null)
```



```

    {
        List<string> list = new List<string>();
        try
            {string sql = "select classroom, c12, c34, c56, c78, c910, c1112 from Date"+time+"
";

                SqlCommand cmd = new SqlCommand(sql, sqlCon);
                SqlDataReader reader = cmd.ExecuteReader();
                while (reader.Read())
                {
                    //将结果集信息添加到返回向量中
                    list.Add(reader[0].ToString());
                    list.Add(reader[1].ToString());
                    list.Add(reader[2].ToString());
                    list.Add(reader[3].ToString());
                    list.Add(reader[4].ToString());
                    list.Add(reader[5].ToString());
                    list.Add(reader[6].ToString());
                }
                reader.Close();
                cmd.Dispose();

            }
        catch(Exception)
        {
            return listnull;
        }
        return list;
    }
}

```

3.3.3 Webservice 项目名称: login

本 Webservice 没有像上面那个一样新建了一个类进行方法调用，而是采用了 LINQ to SQL 类，大大减少了代码量，只需要将数据表直接拖入 LINQ 即可。

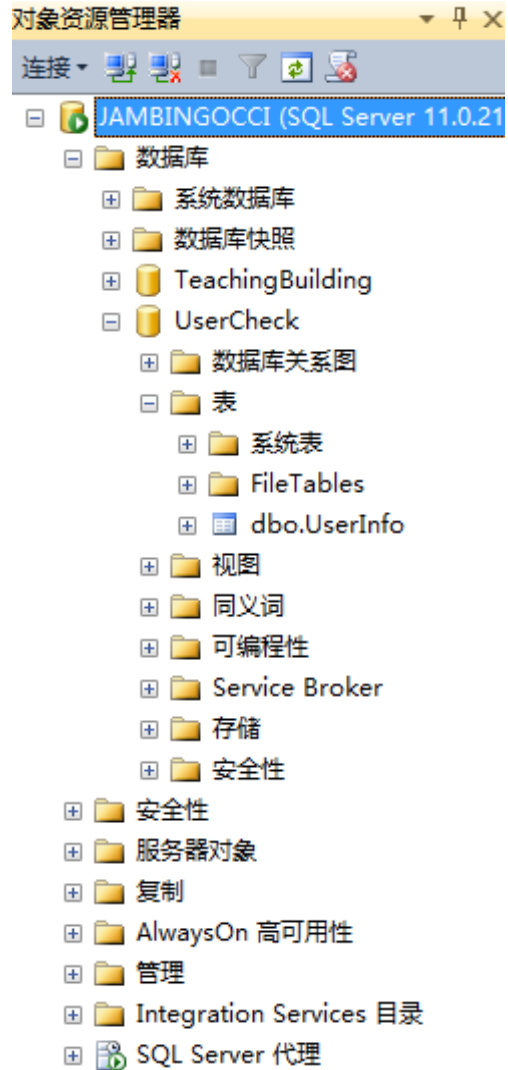
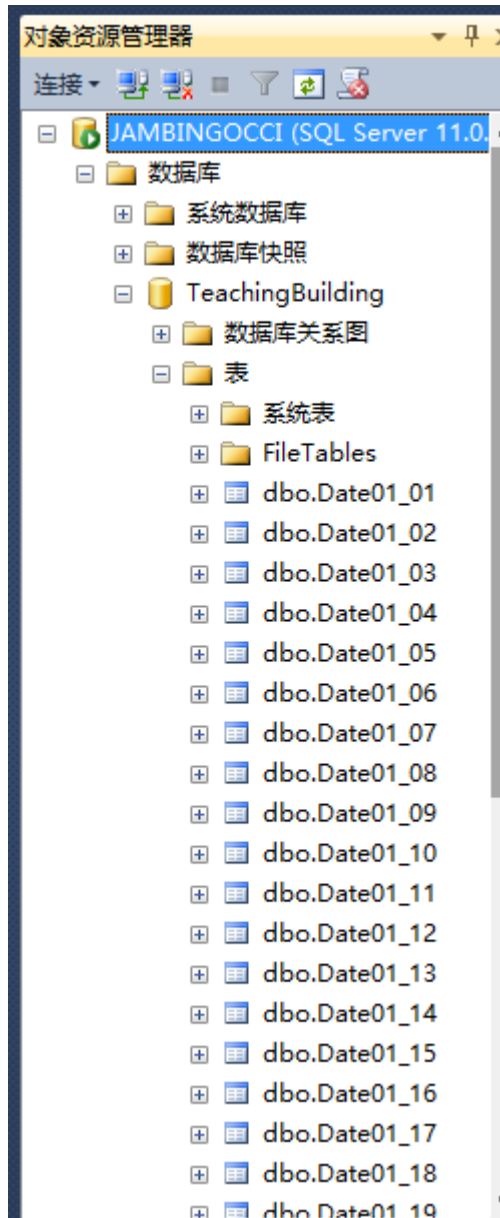
然后再. asmx 文件直接进行方法调用。

WS_Base.asmx:

```

public class WS_Base : System.Web.Services.WebService
{
    public String AuthenticateLogin(string username, string password)///形参为用户名
和登录密码
    {
        using (WS_DataBaseDataDataContext context = new WS_DataBaseDataDataContext())
        {
            var user = from userRow in context.UserInfo
                where userRow.username == username && userRow.password ==
password

```

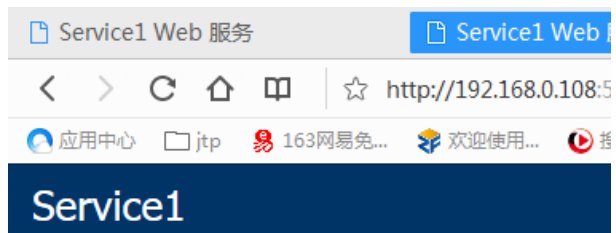
4.2 服务器设计与功能

4.2.1 教室数据库查询服务 (service1 web)

其中方法设计如下：

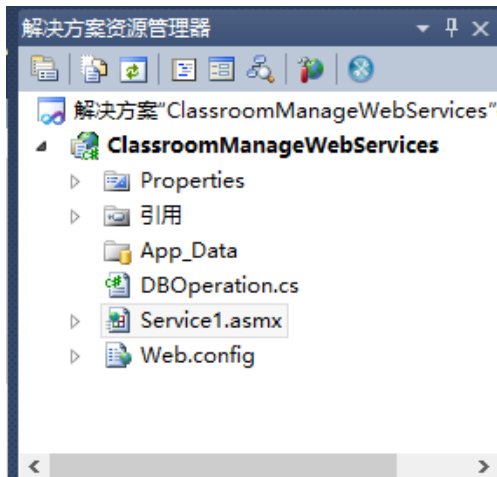
- [HelloWorld](#)
- [searchClassroom1112](#)
11-12 节课信息
- [searchClassroom12](#)
1-2 节课信息
- [searchClassroom34](#)
3-4 节课信息

- [searchClassroom56](#)
5-6 节课信息
- [searchClassroom78](#)
7-8 节课信息
- [searchClassroom910](#)
9-10 节课信息
- [selectAllInfor](#)
获取所有教室的信息



支持下列操作，有关正式定义，请查看[服务说明](#)。

- [HelloWorld](#)
- [searchClassroom1112](#)
11-12节课信息
- [searchClassroom12](#)
1-2节课信息
- [searchClassroom34](#)
3-4节课信息
- [searchClassroom56](#)
5-6节课信息
- [searchClassroom78](#)
7-8节课信息
- [searchClassroom910](#)
9-10节课信息
- [selectAllInfor](#)
获取所有教室的信息



Xml 解析:

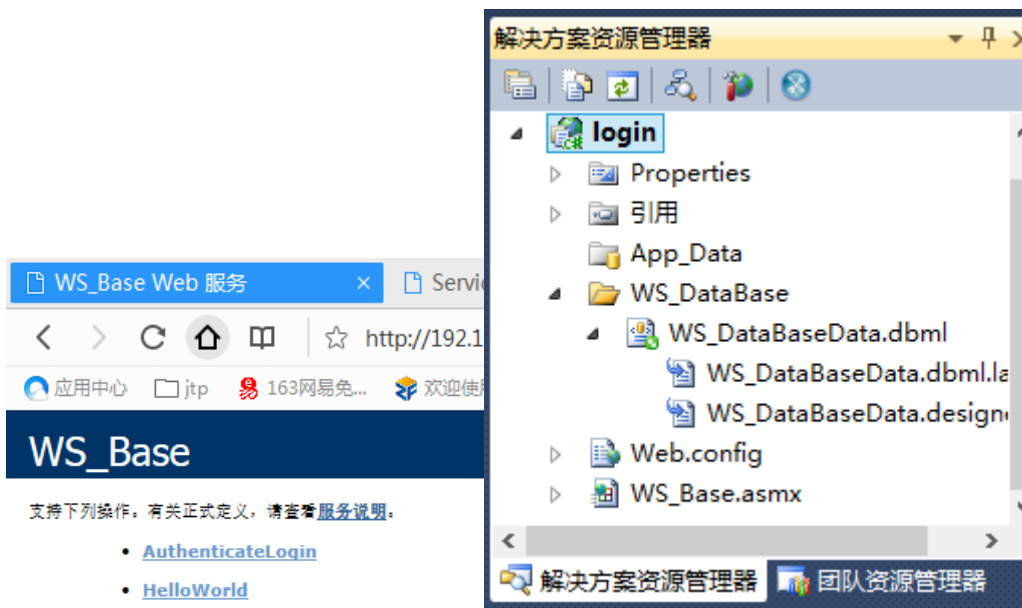


```
<?xml version="1.0" encoding="UTF-8"?>
- <ArrayOfString xmlns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <string>二教105</string>
  <string>二教205</string>
  <string>二教302</string>
  <string>二教304</string>
  <string>二教305</string>
  <string>三教A104</string>
  <string>三教A205</string>
  <string>三教A304</string>
  <string>三教B102</string>
  <string>三教B201</string>
  <string>三教B202</string>
  <string>三教B204</string>
  <string>三教B302</string>
  <string>三教B303</string>
  <string>三教B305</string>
  <string>四教A105</string>
  <string>四教A106</string>
  <string>四教A201</string>
  <string>四教A204</string>
  <string>四教A206</string>
  <string>四教B102</string>
  <string>四教B107</string>
  <string>四教B109</string>
  <string>四教B110</string>
  <string>四教B111</string>
  <string>四教B202</string>
  <string>四教B203</string>
  <string>四教B205</string>
  <string>四教B207</string>
  <string>四教B210</string>
  <string>四教C102</string>
  <string>四教C104</string>
  <string>四教C106</string>
  <string>四教C107</string>
  <string>四教C110</string>
</ArrayOfString>
```

4.2.2 登陆数据库验证服务

其中方法设计如下：

- [AuthenticateLogin](#)
- [HelloWorld](#)



WS_Base

单击[此处](#)，获取完整的操作列表。

AuthenticateLogin

测试

若要使用 HTTP POST 协议对操作进行测试，请单击“调用”按钮。

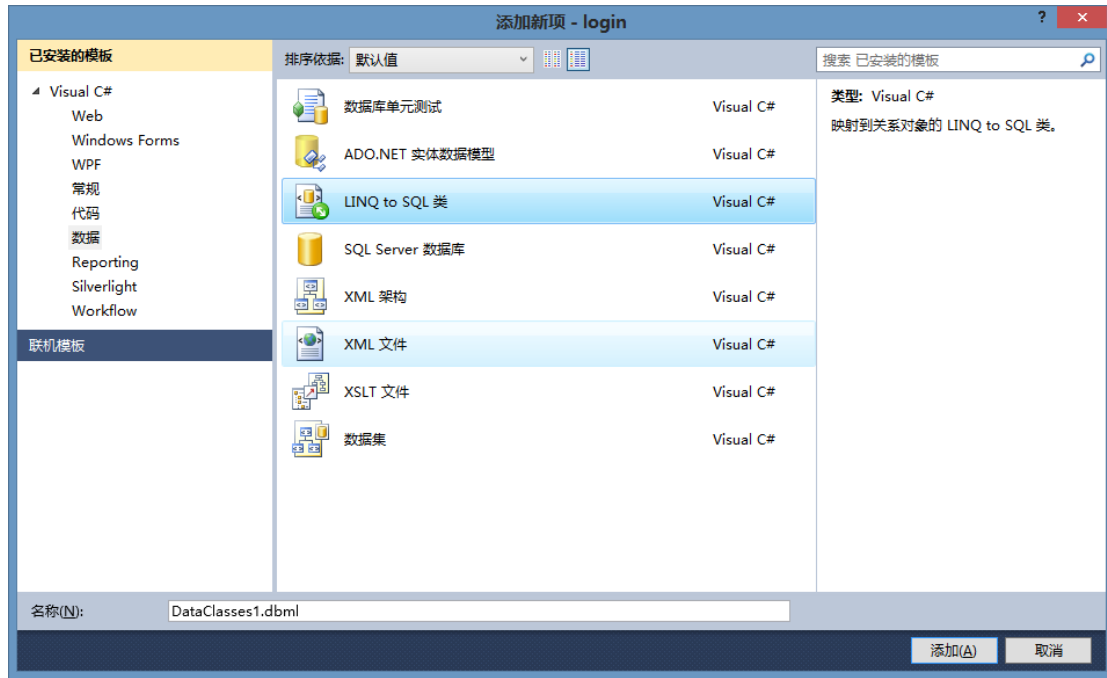
参数	值
username:	<input type="text" value="admin"/>
password:	<input type="text" value="1234"/>

SOAP 1.1

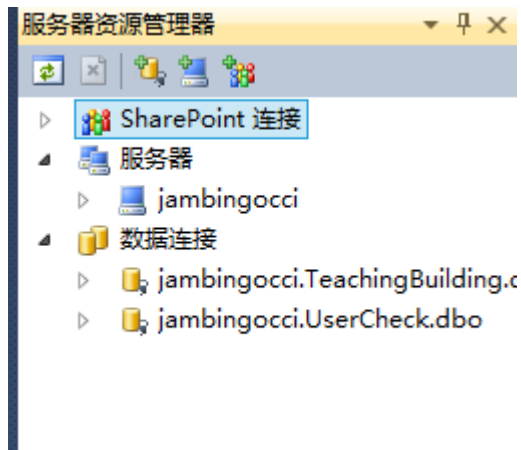
登陆判断：（我设置的返回值为 String 格式的“true”和“false”，方便 soap 解析）

```
<?xml version="1.0" encoding="UTF-8"?>  
<string xmlns="http://tempuri.org/">true</string>
```

LINQ 新建项:



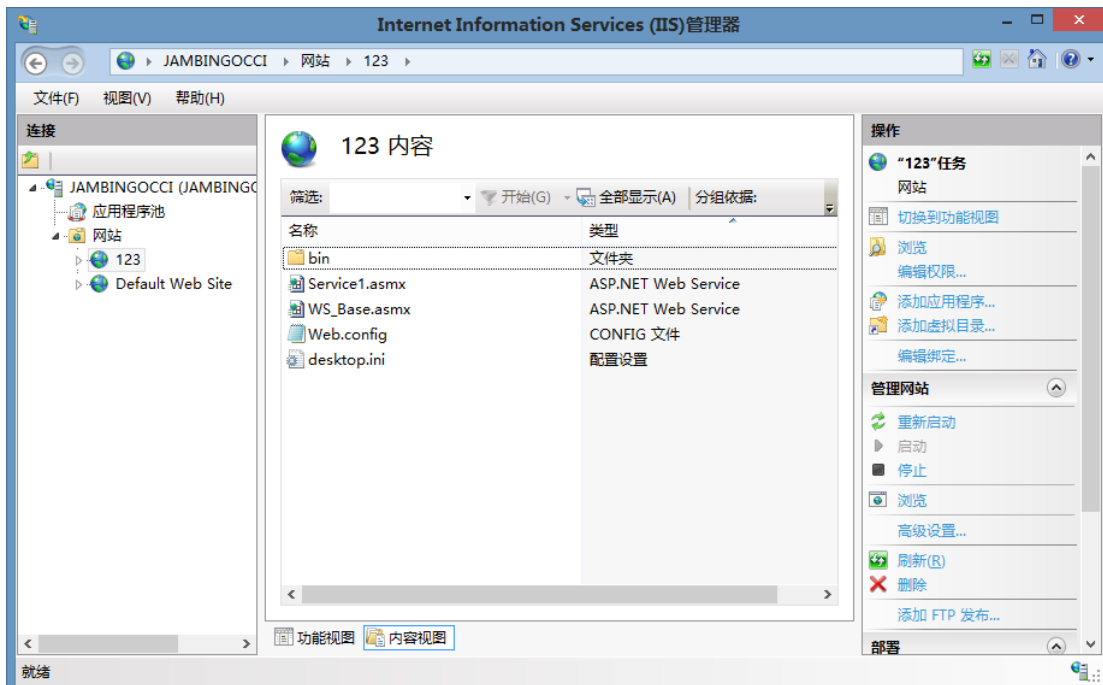
4.2.3 Webservice 连接 sql server 数据库:



4.3.4 Webservice 发布



4.3 IIS 设计



4.4 Android app 设计与功能

登陆界面的输入控制与登录验证:

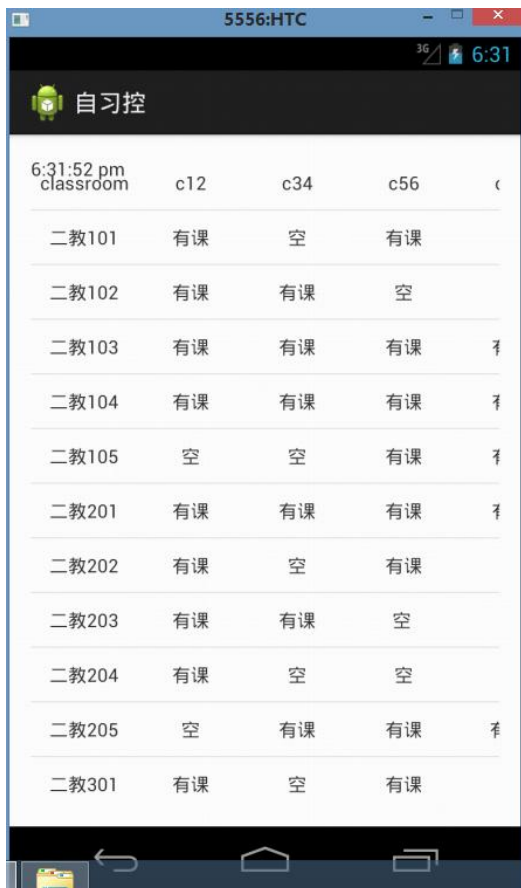
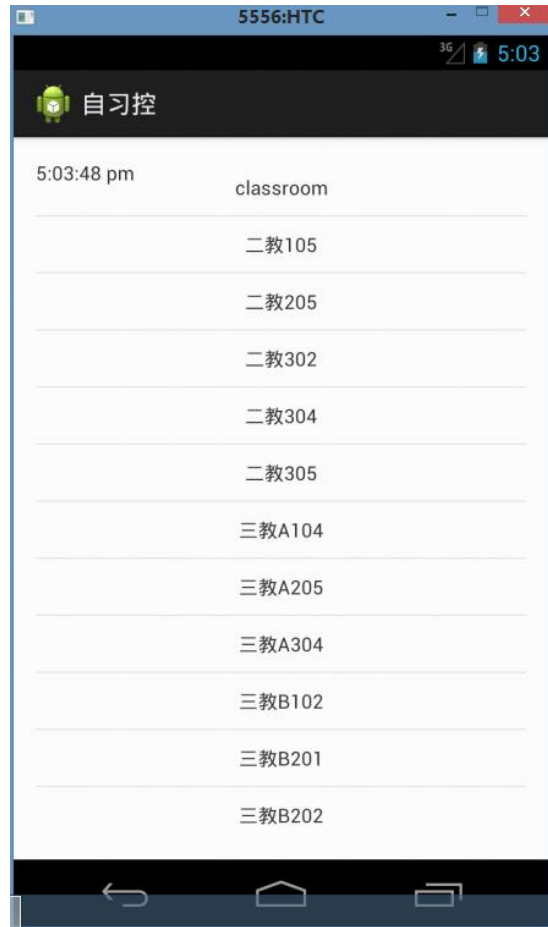
一下 5 图分别为：输入条件有问题、密码数据库验证错误、当前网络无法连接 WEBSERVICE、验证后登陆成功、连接时进度。







主界面（frontpage）：



五、系统测试方案

5.1 黑盒法测试:

案例 1:

软件崩溃:



崩溃情况说明：在点击 1-2,3-4 节课时崩溃，而其他 button 无报错。

报错原因：空指针异常（`nullpointerexception`）。

错误推测（猜错法）：

根据先前程序设计过程中经验，遇到过这个异常，猜测是数据库连接有问题，于是检查服务器情况。

检查 webservice 服务器：

5-6 节课服务器返回 xml 如下：

`Version:1.0 StartHTML:000000217 EndHTML:000003179 StartFragment:000000319`

`EndFragment:000003145 StartSelection:000000319 EndSelection:000003145`

`SourceURL:http://localhost:45290/Service1.asmx/searchClassroom56`

`<?xml version="1.0" encoding="UTF-8"?>`

`<ArrayOfString xmlns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"><string>二教 101 </string><string>三教 A101 </string></ArrayOfString>`

显示连接数据库成功；

1-2 节课服务器返回如下：

`Version:1.0 StartHTML:000000217 EndHTML:000002304 StartFragment:000000319`

`EndFragment:000002270 StartSelection:000000319 EndSelection:000002270`

`SourceURL:http://localhost:45290/Service1.asmx/searchClassroom12`

`<?xml version="1.0" encoding="UTF-8"?>`

`<ArrayOfString xmlns="http://tempuri.org/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>`

意为数据库或读取失败。

进一步猜测为数据库字段或规范问题。检查 aqlserver 数据库：

	classroom	c12	c34	c56	c78	c910	c1112
▶	二教101	有课	有课	空	空	有课	有课
	三教A101	有课	有课	空	空	空	空
	四教A101	有课	有课	有课	有课	空	空
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

发现：数据库中 c12 c34 均不含有“空”字段，使得 webservice 向程序返回 null，soap 无法解析导致程序崩溃。

此例测试了在程序设计过程中没有对 null 返回做判断而导致 app 闪退。

修改如下：

DBOperation.cs

创建 dbnull 方法:

```
public List<string> dbnull() {  
    List<string> lstnull = new List<string>() { "sorry", "database", "empty" };  
    return lstnull;  
}
```

Service1.asmx

if 语句进行判断:

```
public string[] searchClassroom12()  
{  
    if (dbOperation.searchClassroom12().ToArray().equals(null))  
        return dbOperation.dbnull().ToArray();  
    else  
        return dbOperation.searchClassroom12().ToArray();  
}
```

并且在此例测试异常的基础上, 引申将每一个数据库调用函数添加异常 catch:

```
catch (Exception)  
{  
    List<string> listnull = new List<string>() { "数据库发生错误=" };  
    return listnull;  
}
```

以提醒用户, 避免 app 崩溃, 提高程序健壮性与友好性。

案例 2:

数据不能正常显示。

边界值分析:

我们将记录数增加至 100 条以上时, 发现在查询所有信息下出现了无法正常显示数据的情况。此时界面如下。说明能够正常连接 Webservice 与数据库, 并且我们再 log 日志中发现获取了值。但是没能显示的原因我们仍然没有找到。

错误推测:

- 1.可能是 TextView 无法接收超过一定数量的数据。
 - 2.可能采用的 adapter 适配器方式无法接收超过一定数量的数据。
- 由于时间原因, 我们很遗憾没能修正这个 bug。



5.2 测试概要

项目测试从 2015 年 1 月 3 日开始到 2015 年 1 月 6 日结束，共持续 3 天，测试功能点 13 个，执行 20 个测试用例，平均每个功能点执行测试用例 1.5 个，测试共发现 27 个 bug，其中严重级别的 bug 6 个，无效 bug 3 个，平均每个测试功能点 2 个 bug。

5.2.1 测试执行

此次测试严格按照项目计划和测试计划执行，按时完成了测试计划规定的测试对象的测试。针对测试计划规定的测试策略，在测试执行中都有体现，在测试执行过程中，依据测试计划和测试用例，对系统进行了完整的测试。

5.2.2 测试用例

功能性

自习控 app 基本实现了登陆与教室查询的基本功能，达到前台与后台的有序正常运作。

易用性

操作按钮提示信息正确性，一致性，可理解性
限制条件提示信息正确性，一致性，可理解性
必填项标识
输入方式可理解性
中文界面下数据语言与界面语言的一致性

5.3 测试环境

5.3.1 软硬件环境

硬件环境	应用服务器	数据库服务器	客户端
硬件配置	CPU: Intel(R) Celeron(R) CPU 2.40GHz stepping 01 Memory: 1048256k HD: ST380817AS 80G SATA	CPU : Intel(R) Celeron(R) CPU 2.40GHz stepping 01 Memory: 1048256k HD: ST380817AS 80G SATA	CPU: Intel(R) Celeron(R) CPU 2.40GHz stepping 01 Memory: 1048256k HD: ST380817AS 80G SATA
软件配置	OS: CentOS 4.2 JDK 1.5.0_06 Apache 2.2.0 Tomcat 5.5.15	OS: CentOS 4.2 MySQL 5.0.17 Linux	Window 2000 Professional (SP2) IE6.0.2900.2180.xpsp_sp2
网络环境	10M LAN	10M LAN	10M LAN

5.4 测试结果

测试发现的 bug 主要集中在 normal 和 minor 阶段，属于一般性的缺陷，但是测试的时候，出现了 6 个严重级别的 bug，出现严重级别的 bug 主要表现在以下几个方面

系统主要功能没有实现

添加数据代码重复后，出现的找不到页面的错误

多语言处理，未考虑非语种代码的情况

数据库设计未考虑系统管理员角色，导致用系统管理员进行操作的时候出现找不到
页面错误
权限控制异常

严重级别的 bug 主要表现为权限权限 bug:

具有相应按钮操作的权限，页面无相应按钮，无法执行该功能

无相应按钮操作权限，页面有相应按钮，点击按钮能出现权限异常错误

有相应按钮操作权限，有相应按钮，执行该功能出现权限异常错误