

分类号: TP311.1

U D C: D10621-408-(2015) 2407-0

密 级: 公 开

编 号: 2011124039

成 都 信 息 工 程 大 学

学 位 论 文

基于 Windows 的班级管理应用的设计与实现

论文作者姓名: 万俊嘉

申请学位专业: 物联网工程

申请学位类别: 工科学士

指导教师姓名(职称): 昌燕 (副教授)

论文提交日期: 2015 年 06 月 01 日

摘 要

班级管理应用是面向学校和班级对班级信息进行有效管理的一个基于 Windows 的通用应用。开发该应用是为了实现班级信息管理的移动化管理。随着移动互联网的高速发展以及智能设备的小型化、移动化，对信息的管理，信息的传达有了更具时效和更有效率的方式，班级管理应用由此而生。该应用基于 JavaScript 语言，借助微软的全新理念：UWP(Universal Windows Platform)，并且使用了 Sql Server 数据库对信息的进行了有效的管理和存储。该应用可安装在 Windows 8 以上版本应用的电脑或者 Windows Phone 手机上，班委和学生可根据各自账号登陆该应用，了解班级的相关信息，并且发送通知信息和组织活动。最后，在 UWP 框架的帮助下，实现了 PC 端和移动端的统一体验。

关键词：班级管理；移动互联网；移动化；JavaScript；UWP

ETD 论文管理系统

The Design and Implementation of a Class Information Management APP Windows-based

Abstract

The more effective and flexible mobile business system has become feasible as the consequence of the developed mobile smart phone, the miniaturized version of computer and the mobile network of high bandwidth. Under this circumstance, we had developed a Class Information Management system for mobile class management on Windows platform, including Windows phone, desktop and tablet. The application is built with JavaScript Language and based on the Universal Windows Platform (UWP) application framework and the SQL-Server database technology. It provides students and their teachers both for interaction and information delivery and sharing such as class collective activities, sharing of class notes and etc. to improve the teaching performance. The last point is that due to the UWP, the application will deliver a uniform user experience across every Windows device.

Key words: class information management, the mobile Internet, JavaScript, UWP

目 录

论文总页数：33 页

1.1 课题背景	1
1.2 国内外研究现状	1
2 总体设计和需求分析	1
2.1 需求分析	1
2.1.1 业务需求分析	1
2.1.2 功能需求分析	1
2.2 可行性分析	2
2.2.1 技术可行性分析	2
2.2.2 经济可行性分析	2
2.2.3 操作可行性分析	2
3 基本技术介绍	2
3.1 JAVASCRIPT 技术	3
3.2 WINJS (WINDOWS LIBRARY FOR JAVASCRIPT)	3
3.3 微软 UWP 开发平台	3
3.4 ASP.NET WEB API 简介	4
3.5 HTTP 协议	4
4 系统模块设计	4
5 系统详细设计	5
5.1 概念模型设计	5
5.2 数据库设计	5
5.3 数据库 ER 图设计	10
6 功能实现	10
6.1 服务端基本功能实现	10
6.1.1 配置文件	10
6.1.2 访问路由配置	10
6.1.3 GET 请求处理	11
6.1.4 PUT 请求处理	11
6.1.5 POST 请求处理	12
6.1.6 DELETE 请求处理	13
6.2 应用功能实现	13
6.2.1 应用流程描述	13
6.2.2 相关配置	16
6.2.3 数据通信	16

6.2.4 页面控件的实现.....	18
6.2.5 页面实现效果.....	19
7 调试与测试.....	27
7.1 运行环境.....	27
7.2 测试原则.....	27
7.3 测试内容.....	27
7.3.1 网络检测测试.....	27
7.3.2 登录提示测试.....	28
7.3.3 输入判空测试.....	28
7.3.4 数据通信测试.....	29
7.4 常见问题和解决方法.....	29
8 改进建议和措施.....	29
结 语.....	30
参考文献.....	31
致 谢.....	32
声 明.....	33

ETD 论文管理系统

1 引言

1.1 课题背景

自智能移动设备出现以来，移动互联网获得了极大的发展，移动互联网的高速发展，反过来又促进了智能移动设备的不断发展。智能移动设备的普及率，在各个大学校园中，几乎已是 100%。这就为我们管理班级信息提供了更加方便和更加直接有效的方式。因此，通过基于移动互联网的班级管理应用，就可以做到随时随地的查看和管理，及时获得相关的通知信息，极大的方便老师和同学们。

1.2 国内外研究现状

目前，随着移动设备和移动互联网的不断发展，移动办公已经成为了一种正在实现的趋势，正如微软的移动版 Office 办公软件，以及谷歌的移动版办公软件一样，越来越多的公司和机构都把工作场景向移动端延展。然而，就学校来说，信息管理的移动化却是远远不够的。现在学校的信息管理方式还是基于 PC 端，并且技术落后，很多时候当我们不方便使用 PC 进行访问时，只能使用随身携带的移动设备进行，然而学校却并没有对此做好准备，因此，目前来说，学校的移动化管理做的远远不足。因此，一个可以对学校信息进行移动化管理的应用必不可少。

2 总体设计和需求分析

2.1 需求分析

2.1.1 业务需求分析

该应用主要使用者是老师和学生，班级学生信息查询、课程查询、班费管理以及发送实时通知和活动信息为主要业务。随着大学生生活和学习的方式不断的改变，当前大学生总体来说是分散的，除了上课和某些会议以及活动会聚到一起之外，学生们很少聚到一起。另外，目前辅导员或者班长通知信息主要是在 QQ 群里发送信息通知为主，不过由于 QQ 群信息刷新过快，再加上并不是所有人都会及时去看公告信息，导致信息不能及时传达。因此，信息的实时通知非常重要。

2.1.2 功能需求分析

业务需求主要是面向业务人员，功能需求面向使用者。

经过调查以及分析，目前该应用应具备如下功能：

具备二级权限：

班委：可查看本班班级基本信息，本班班费及班费使用情况，学生信息，班级活动信息，课程信息，发送通知，以及修改以上信息；

学生：可查看以上提到的所有信息，仅能编辑本人信息，以及课程信息。

班级基本信息展示，包括：学院、专业、年级、辅导员、班委等信息；

班费总览信息将以饼状图展示，包括：班费总额、当前金额、班费收集时间、班费保管人、第几次收集班费等信息。所有人都可以清楚的了解到班费情况；

班费详情信息展示，所有人可以从这里详细的了解班费是谁用了，什么时候用了，用到哪里去了；

班级活动信息发布与展示，班长可以发布最新活动，所有人可以在这里获取班级最新活动信息，并统计愿意参加该活动的名单；

通知信息发布与展示，辅导员或者班长可以发布通知信息，所有人可以查看通知信息并发回已查阅信息；

课程信息的编辑与展示，课程可加锁，加锁后任何人均不可修改课程信息，解锁后可制定哪些人可修改课程信息；

学生信息编辑与展示，每个人仅能修改自己的相关信息，所有人的信息合在一起，就是完整的学生名单。

2.2 可行性分析

2.2.1 技术可行性分析

该应用采用 JavaScript 语言编程技术并基于微软 UWP 开发框架开发。主要采用 Visual Studio 2015 RC 软件工具和 Sql Server 2014 数据库对网站进行开发和维护。从现有的技术条件来说，开发工具、技术框架以及开发人员都已经准备好，因此，从技术上来说，此应用的开发具备可行性。

2.2.2 经济可行性分析

从长远来看，信息的移动化管理与处理是必然的趋势，因此，越是早进行移动化部署，越是节约成本。该应用基于 Windows 系统开发，采用 UWP 应用架构，实现了一次开发，到处部署(Windows 设备)，有着出色的统一体验，并且，移动化的信息管理与处理，能极大的提高对信息的管理效率，使得信息不会滞后，为此节约了时间成本和人力成本。同时，开发软件和语言都是免费的，也不用为此付出额外的费用。所以，从经济上来看，该应用具备可行性。

2.2.3 操作可行性分析

因为该应用基于 UWP 架构，使其可以安装在 Windows 8 及其以上版本系统的电脑或者平板以及 Windows Phone 手机上，并且提供了统一的视觉体验以及出色的触控体验。应用整体风格整洁，容易理解和操作，信息能够得到及时的处理，使得每个人看到的信息基本一致，并且某些操作通过这种移动化的处理，将会大大提高对班级的管理效率。因此，从操作上来看，该应用具备可行性。

3 基本技术介绍

3.1 JavaScript 技术

JavaScript 语言（以上简称 JS），是一个面向对象的、轻量级的、解释型的、以及将函数视为一等公民的编程语言。JS 最初只是一门得不到重视的脚本语言，然而随着移动互联网和移动智能设备的爆发式发展，JS 大放异彩。各种各样的 JS 库如雨后春笋一般冒了出来，使得开发者能够快速并且优雅的实现各种有趣的页面以及应用。除了 web 端有着出色的表现，JS 在服务器端一样有着非常出色的表现，如：node.js。除此之外，微软将 JS 引入自家的应用开发体系，使得 web 开发者能够轻松的开发微软通用应用。

3.2 WinJS (Windows Library for JavaScript)

WinJS 是微软推出的一个 JavaScript 库，通过 WinJS，我们可以采用 HTML、CSS 和 JavaScript 构建有着统一体验的通用应用，还可以用于 WEB 开发，以及使用基于 HTML 的应用技术（如 Apache Cordova）。WinJS 提供了高质量的基础架构（页面控件、数据绑定等），精美的 UI 功能（如虚拟化集合等）和高性能的 Windows 控件。现在，WinJS 稳定版本为 3.0，并且发布了 WinJS4.0 的预览版，提供了如下特性：

- 使你的应用在不同的设备上有着统一的体验；
- 能够和别的 JavaScript 库很好的一起工作；
- 更多的控件。

3.3 微软 UWP 开发平台

微软 UWP 给我们提供了以下 4 点特性：

- 一种新的构建 Windows apps 方式；
- 跨平台，跨设备；
- 新的控件和面板让你的 UI 适配不同的设备；
- 新的 API 和新的操作方式。

UWP(Universal Windows Platform)提供了跨设备的稳定的 API 层，我们可以创建并安装应用到许多的设备上。如图 1。

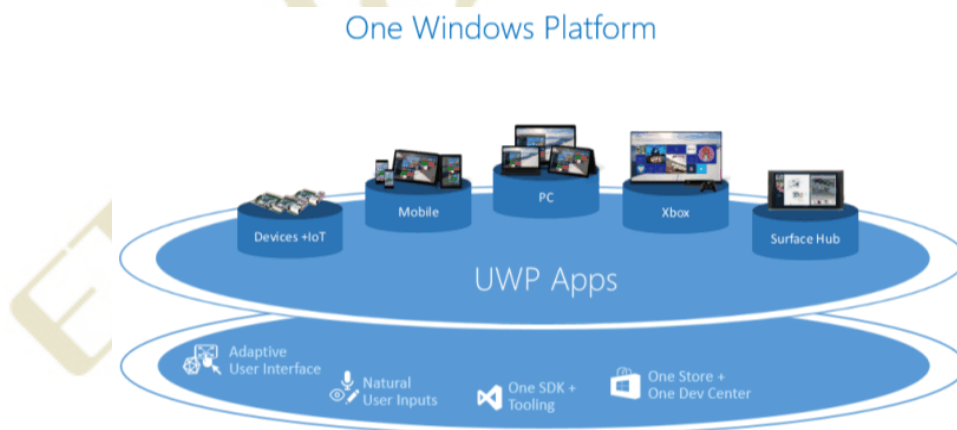


图 1. One Windows Platform

当 UWP apps 运行在许多有着不同形式和输入方式的设备上时，可以完美的适配这些设备并且发挥出它们的最好性能。在 UWP 平台上，可以编写代码来有条件地访问这些独特的 API，从而使应用在其他设备上呈现不同的体验。响应式的控件和新的布局面板将帮助我们的应用在不同的屏幕和分辨率下仍然表现的十分完美。

3.4 ASP.NET Web API 简介

这是一个可以为包括浏览器、移动设备等多种客户端提供 HTTP 服务的新框架，ASP.NET Web API 是构建 RESTful 服务的理想平台。

该技术包含以下特性：

- 先进的 HTTP 编程模型；
- 支持路由；
- 内容协商；
- 模型绑定与验证；
- 过滤；
- 查询聚合；
- 改进的 HTTP 细节可测性；
- 改进的依赖反转（IOC）支持；
- 基于代码的配置；
- 自托管。

3.5 HTTP 协议

HTTP（Hypertext Transfer Protocol）协议，是用于从 WWW 服务器传输超文本到本地浏览器的传送协议，它可以是浏览器更加高效，使网络传输减少。HTTP 协议是一个应用层协议，有请求和相应构成，是一个标准的客户端服务器模型。HTTP 是一个无状态协议。如图 2



图 2. HTTP 协议简易模型

4 系统模块设计

本程序主要分班委，学生三个权限。班委可查看本班的信息，发送通知和活动信息，可修改学生、班级、班费等信息；学生可查看本班班级信息，活动信息，班费信息，课程信息并编辑自己的学生信息，以及相关的课程信息。总功能模块图如图 3 所示。

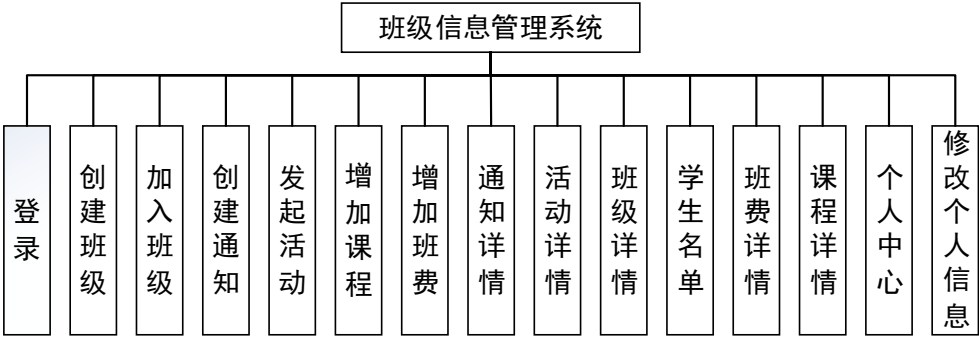


图 3 系统模块图

5 系统详细设计

5.1 概念模型设计

这里采用 E-R 图（实体-联系模型）的方式来表现概念模型的设计，通过实体、联系、属性这三个基本成分直观准确的从概念上表示了数据库信息组织形式。系统 E-R 图如图 4。

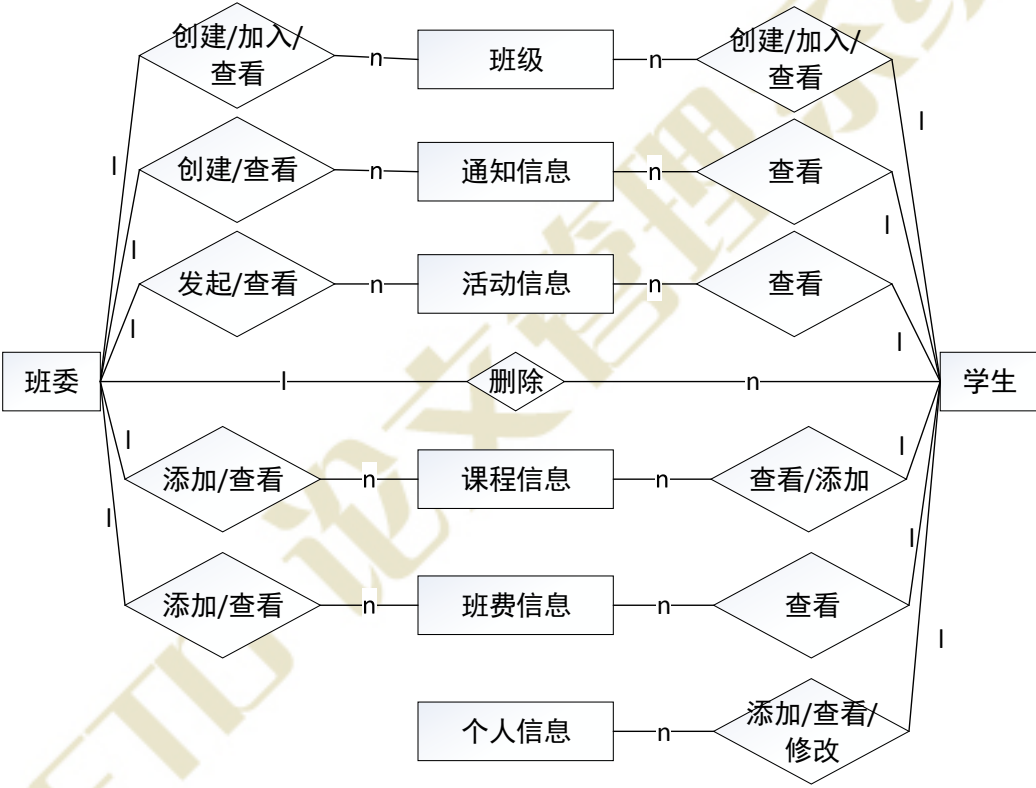


图 4. 系统 ER 图

5.2 数据库设计

1. 班级信息表（班级 ID、学院、年级、班级号、专业、辅导员、班长、副

班长、团委、生活委员、总人数、当前人数、男生人数、女生人数、备注), 如表 1:

表 1. 班级信息表

字段	类型	Null	说明
班级 ID(ClassID)	nvarchar(10)	Y	主键 ID, 由年级 + 班级 + 时间戳生成
专业(major)	Nvarchar (50)	Y	所读专业名字
年级(grade)	Nvarchar (8)	Y	入学时间既年级
班级号 (classNum)	Nvarchar (8)	Y	同专业下第几个班级, 如 01 班班级号为 01
学院(college)	Nvarchar (50)	Y	所属学院
辅导员 (counselor)	Nvarchar (20)	Y	辅导员的姓名
班长(monitor)	Nvarchar (20)	Y	班长的姓名
副班长 (viceMonitor)	Nvarchar (20)	N	副班长的姓名
团委 (CYLCommittee)	Nvarchar (20)	N	团委的姓名
生活委员 (treasurer)	Nvarchar (20)	N	生活委员的姓名
总人数 (totalNum)	int	N	班级应该有的总人数
当前人数 (currentNum)	int	N	当前一共有多少人在校
男生人数(male)	int	N	男生的人数
女生人数 (female)	int	N	女生的人数
备注(note)	Nvarchar (200)	N	班级备注信息

2. 学生信息表 (学号、密码、性别、姓名、邮箱、电话、宿舍楼、寝室号、是否在校、角色、备注), 如表 2:

表 2. 学生信息表

字段	类别	Null	说明
学号(stuID)	Nvarchar(20)	Y	学号, 作为主键
密码(password)	Nvarchar(20)	Y	登陆密码, 默认为学号
姓名(name)	Nvarchar(20)	Y	学生名字
性别(sex)	int	Y	状态码:1: 男 2: 女
电话(telephone)	Nvarchar(20)	N	联系电话

邮箱(email)	Nvarchar(50)	Y	常用邮箱
宿舍号(hallNum)	Nvarchar(20)	Y	所住宿舍楼栋数
寝室号(chamberNum)	Nvarchar(20)	Y	寝室号码
是否在校(isSchool)	int	Y	状态码: 1:正常在校 2:休学 3:请假
角色(role)	int	Y	状态码: 1 学生 / 2 班委
备注(note)	Nvarchar(200)	N	学生备注信息

3. 班级活动信息表（活动 ID、活动名、活动内容、活动开始时间、活动结束时间、活动地点、活动参与人数、活动参与名单、备注），如表 3：

表 3. 班级活动信息表

字段	类别	null	说明
活动 ID(activityID)	Nvarchar	Y	活动 ID，主键
活动名(activityName)	Nvarchar	Y	本次活动的名字
活动内容(activityContent)	Nvarchar	Y	本次活动的内容
活动开始时间(activityStartTime)	Datetime	Y	本次活动开始时间
活动结束时间(activityEndTime)	Datetime	Y	本次活动结束时间
活动地点(activityPlace)	Nvarchar	Y	本次活动所在地点
活动参与人数(activityPersonNum)	Int	N	本次活动参与的人数
活动参与名单(activityContributors)	Nvarchar(4000)	N	本次活动参与者的名单
备注(note)	Nvarchar(200)	N	本次活动备注信息

4. 班费总表（班费 ID、班费收集次数、班费总额、班费收集时间、班费保管人、备注），如表 4：

表 4. 班费总表

字段	类别	Null	说明
班费 ID(classFundID)	int	Y	班费 ID,主键
班费总额(classFund)	Decimal(10,2)	Y	班费剩余总额
班费收集次数(times)	int	N	第几次收集班费

班费收集时间 (collectTime)	dateTime	Y	收集班费的时间
班费保管人 (fundKeeper)	Nvarchar(20)	Y	保管班费的人
备注	Nvarchar(200)	N	班费备注

5. 花费详情表（花费项目 ID、花费项目名字、花费金额、结算人、花费时间、备注），如表 5：

表 5. 花费详情表

字段	类别	Null	说明
花费项目 ID(spendItemID)	Nvarchar(20)	Y	花费项目 ID，主键
花费项目类型 (spendItemType)	int	Y	花费到某个方面，比如活动，奖金
花费项目名 (spendItemName)	Nvarchar(20)	Y	花费项目名字
花费数目 (spendCount)	Decimal(10,2)	Y	花费了多少钱
结算人 (spendPerson)	Nvarchar(20)	Y	结算这笔花费的人
结算时间 (spendTime)	dateTime	Y	结算这笔花费的时间
备注(note)	Nvarchar(200)	N	这笔花费的备注信息

6. 通知信息表（通知 ID、通知标题、通知内容、通知人、通知时间、重要等级、关键字、备注），如表 6：

表 6. 通知信息表

字段	类别	Null	说明
通知 ID(noticeID)	Nvarchar(20)	Y	通知 ID,主键
通知标题(title)	Nvarchar(100)	Y	通知标题
通知内容 (content)	Nvarchar(500)	Y	通知内容
通知人(person)	Nvarchar(20)	Y	发出通知的人
发出通知时间 (time)	Datetime	Y	发出通知具体时间
重要等级(level)	Int	Y	状态码：1:很重要 2：重要 3：一般
关键字(keyword)	Nvarchar(50)	Y	通知关键字
备注(note)	Nvarchar(200)	N	通知备注

7. 课程信息表（课程 ID、课程名、上课老师、上课老师电话、上课老师邮箱、上课教室、上课开始周、上课结束周、上课时间、是否有实验课、实验课老师、实验课老师电话、实验课教室、实验课开始周、实验课结束周、实验课时间、是否必修、课程类型、课程持续周数、考试时间、考试方式、备注），如表 7：

表 7. 课程信息表

字段	类别	null	说明
课程 ID(courseID)	Nvarchar(20)	Y	课程 ID，主键
课程名 (courseName)	Nvarchar(50)	Y	课程名字
授课老师 (teacher)	Nvarchar(20)	Y	上课老师名字
电话(tel)	Nvarchar(20)	Y	上课老师的电话
邮箱(email)	Nvarchar(50)	N	上课老师的邮箱
授课地点 (classroom)	Nvarchar(20)	N	上课的教室，可能有多个
是否有实验 (isHaveLab)	Int	Y	状态码：1:有 2:没有
实验开始时间 (labStartTime)	Int	N	实验课第几周开始
实验结束时间 (labEndTime)	Int	N	实验课第几周结束
实验周数 (labWeeks)	Int	N	实验课持续周数
实验时间 (labTime)	Datetime	N	实验课上课时间
上课时间 (classTime)	Datetime	Y	上课时间
开课时间 (startTime)	Date	Y	第几周开课
结课时间 (endTime)	Date	Y	第几周结课
是否必修 (isRequired)	Int	Y	状态码：1:必修 2:选修
课程类型(type)	Int	Y	状态码：1:基础课程 2:专业课程
课程周数 (courseWeeks)	Int	N	该课程持续周数
考试时间	Date	N	考试时间

(testTime)			
考试方式 (testMode)	Int	N	状态码: 1:开卷 2:闭卷 3:机 考
备注(note)	Nvarchar(200)	N	课程备注

5.3 数据库 ER 图设计

在这里，采用 ER Studio 实现了数据库的 ER 图，通过 ER 图，我们可以清晰直观的看出表与表之间的关系，以便于之后的开发。数据库 ER 图如图 5。

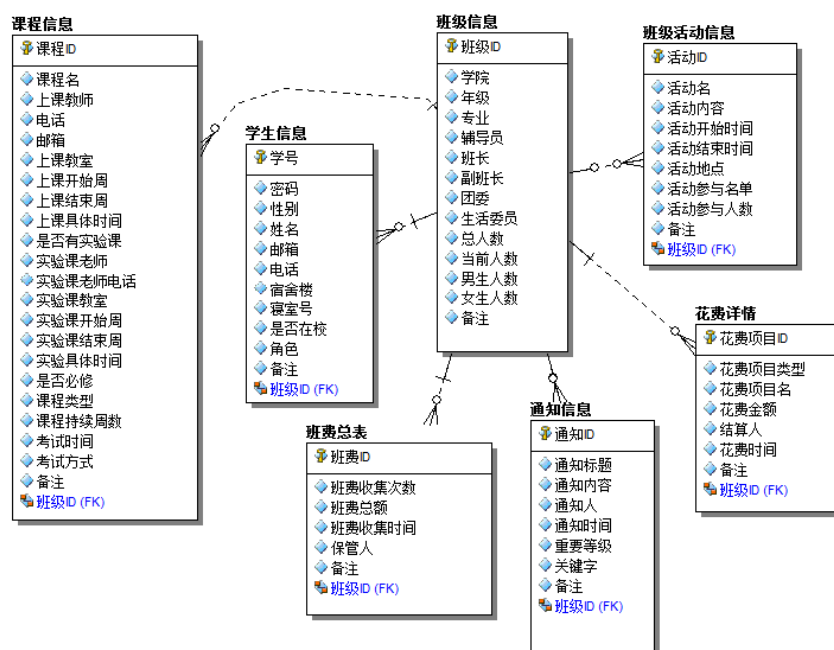


图 5. 数据库 ER 图

6 功能实现

6.1 服务端基本功能实现

6.1.1 配置文件

数据库连接配置：

```
<connectionStrings>
  <add name="IConnectString" connectionString="data source=localhost;initial
catalog=ClassInfoDB;user
id=sa;password=mysqlserver;MultipleActiveResultSets=True;App=EntityFramework"
providerName="System.Data.SqlClient" />
</connectionStrings>
```

6.1.2 访问路由配置

这里将对请求 URI 的格式进行配置。

```
// Web API 路由
```



```
config.MapHttpAttributeRoutes();
```

```
config.Routes.MapHttpRoute(  
    name: "DefaultApi",  
    routeTemplate: "api/{controller}/{id}",  
    defaults: new { id = RouteParameter.Optional }  
);
```

6.1.3 GET 请求处理

根据 HTTP 协议的定义，GET 请求用于获取 URI 资源，该操作不对服务端有任何影响，这里 GET 请求分两种，一种是基本的查询，获取所有数据；另一种是根据某个或多个字段去查询数据。

- 基本查询：

```
// GET: api/ClassInfo  
public IQueryable<ClassInfo> GetClassInfo () {  
    return db.ClassInfo;  
}
```

- 根据 ID 查询：

```
// GET: api/ClassInfo/5  
[ResponseType(typeof(ClassInfo))]  
public IHttpActionResult GetClassInfo (string id) {  
    ClassInfo classInfo = db.ClassInfo.Find(id);  
    if (classInfo == null) {  
        return NotFound();  
    }  
  
    return Ok(classInfo);  
}
```

6.1.4 PUT 请求处理

根据 HTTP 协议的定义，PUT 请求用于更新 URI 上的一个资源。如果服务端允许，PUT 也可以用来新建一个资源，不过，这个操作建议使用 POST 请求。

```
// PUT: api/ClassInfo/5  
[ResponseType(typeof(void))]  
public IHttpActionResult PutClassInfo (string id, ClassInfo classInfo) {  
    if (!ModelState.IsValid) {  
        return BadRequest(ModelState);  
    }  
  
    db.ClassInfo.Add(classInfo);  
    db.SaveChanges();  
    return StatusCode(HttpStatusCode.NoContent);  
}
```

```

    }

    if (id != classInfo.ClassID) {
        return BadRequest();
    }

    db.Entry(classInfo).State = EntityState.Modified;

    try {
        db.SaveChanges();
    }
    catch (DbUpdateConcurrencyException) {
        if (!ClassInfoExists(id)) {
            return NotFound();
        }
        else {
            throw;
        }
    }
    return StatusCode(HttpStatusCode.NoContent);
}

```

6.1.5 POST 请求处理

根据 HTTP 协议的定义，POST 请求用于新建资源，服务端在指定的 URI 上新建一个对象，将新建资源的地址作为响应资源的一部分返回。

```

// POST: api/ClassInfo
[ResponseType(typeof(ClassInfo))]
public IHttpActionResult PostClassInfo (ClassInfo classInfo) {
    if (!ModelState.IsValid) {
        return BadRequest(ModelState);
    }

    classInfo.TotalNum = classInfo.Male + classInfo.Female;
    classInfo.ClassID = classInfo.Grade + classInfo.ClassNum + GetTimeStamp();

    db.ClassInfo.Add(classInfo);
}

```

```

        try {
            db.SaveChanges();
        }
        catch (DbUpdateException) {
            if (ClassInfoExists(classInfo.ClassID)) {
                return Conflict();
            }
            else {
                throw;
            }
        }

        return CreatedAtRoute("DefaultApi", new { id = classInfo.ClassID }, classInfo);
    }
}

```

6.1.6 DELETE 请求处理

根据 HTTP 协议的定义，DELETE 请求用于删除指定 URI 的资源。

```

// DELETE: api/ClassInfo/5
[ResponseType(typeof(ClassInfo))]
public IHttpActionResult DeleteClassInfo (string id) {
    ClassInfo classInfo = db.ClassInfo.Find(id);
    if (classInfo == null) {
        return NotFound();
    }
    db.ClassInfo.Remove(classInfo);
    db.SaveChanges();
    return Ok(classInfo);
}

```

6.2 应用功能实现

6.2.1 应用流程描述

1. 主流程

未添加过班级的需要新创建班级，闯将班级成功后这个班级的所有人就可以加入班级，加入成功后，就可以登陆系统了；如果已经有班级存在并且加入了班级的可以直接登录系统。若登录失败，需要重新登录。登录成功后，会根据登录账号验证权限，分为所有人都是可以进行的操作以及班委才能进行的操作，完成操作后可结束并退出系统。主流程如图 6。

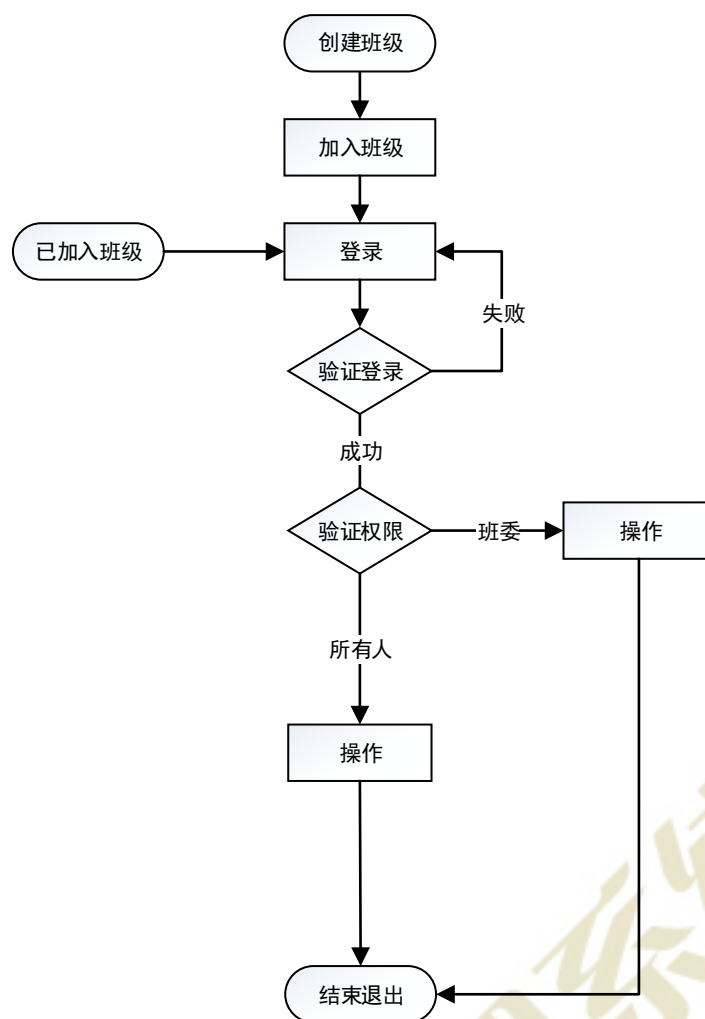


图 6. 主流程图

2. 所有人（包括学生和班委）权限流程

登录并验证权限后，分为所有人权限，即学生和班委以及辅导员都具备的操作权限，以及班委权限。在该权限限制下，可进行查看信息操作，以及修改个人信息、添加班级和添加课程三个操作，查看操作流程如图 7。

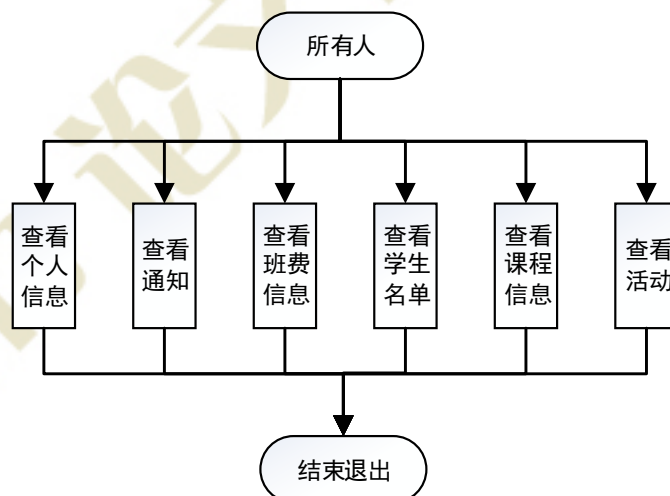


图 7. 所有人查看权限流程图

修改个人信息、添加班级和添加课程三个操作流程如图 8。

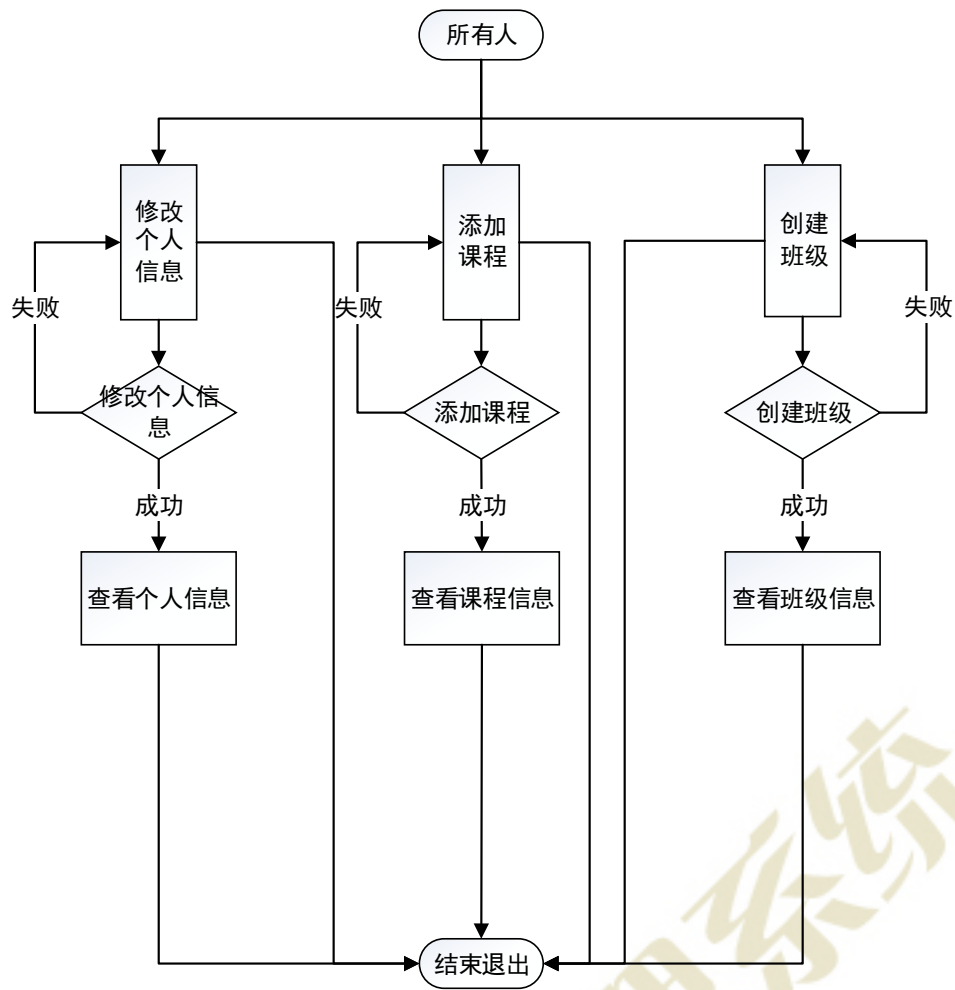


图 8. 所有人修改权限流程图

3. 班委权限流程

在班委权限下，可进行创建通知、增加班费信息、修改班费信息、修改课程和删除学生操作。班委权限流程图如图 9。

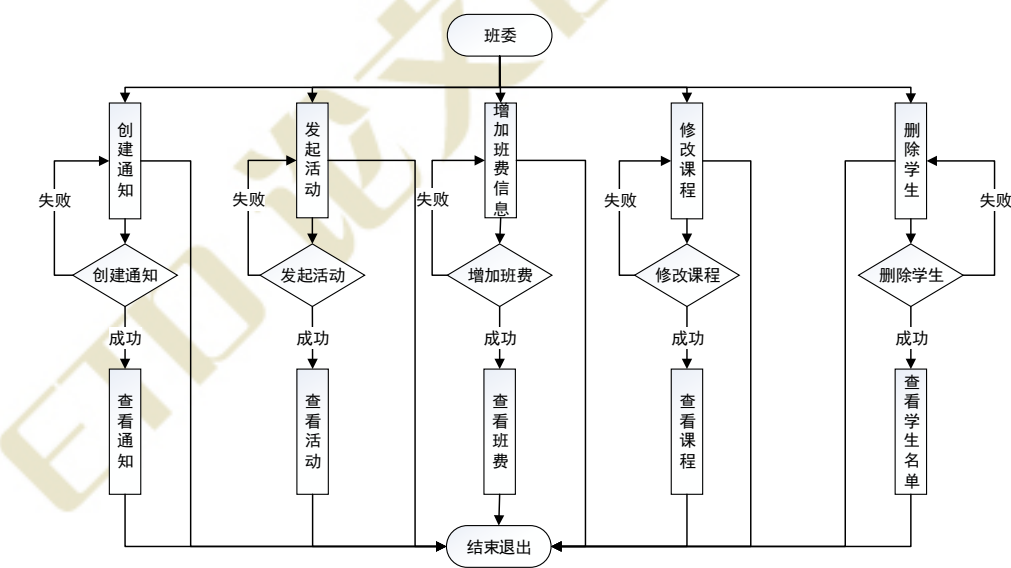


图 9. 班委权限流程图

6.2.2 相关配置

打开 package.appxmanifest 文件，配置相关信息。首先，配置应用名，以及起始页，如图 10。

应用程序

可见资产

要求

功能

声明

内容 URI

打包

使用此页可以设置用于标识和描述应用程序的属性。

显示名称:

班级信息管理系统改进版

起始页:

default.html

默认语言:

zh-CN

[详细信息](#)

说明:

ClassInfoManageSys.WindowsPhone

图 10. 配置应用名

然后配置网络相关，在功能选项面板中选中 Internet (客户端和服务端) 一项，如图 11。

应用程序

可见资产

要求

功能

声明

使用此页可以指定应用程序可使用的系统功能或设备。

功能:

☐ 约会

☐ 联系人

☐ 企业身份验证

☒ Internet (客户端和服务端)

☐ 位置

☐ 麦克风

☐ 音乐库

☐ 图片库

☐ 邻近

☐ 可移动存储

☐ 共享用户证书

☐ 视频库

☐ 网络摄像机

说明:

提供对用户的约会应用商店的访问权
约会应用商店中的其他应用程序，并
并将约会写入所创建的日历中。
[更多信息](#)

图 11. 配置网络功能

建立 url.js 文件配置统一资源定位器：

```
var urlString = 'http://192.168.1.104:13572/api/';
```

6.2.3 数据通信

应用基于 WinJS 框架实现，内部大量使用了 XMLHttpRequest 请求从服务端获取数据，请求方式：

首先将 url.js 文件引入到起始页面：

```
<script type="text/javascript" src="/js/data/url.js"></script>
```

获取请求 url：

```
var urlString = window.urlString;
```

第一种请求方式，使用 WinJS.xhr()：

```
WinJS.xhr({
    url: urlString + "ClassInfo",
    type: 'GET',
    data: data,
    responseType: 'json'
}).done(function () {
    //doSomething
})
```

第二种请求方式，构建一个 XMLHttpRequest 对象：

```
var xhr = new XMLHttpRequest();
```

```
xhr.open("POST", urlString + "ClassInfo", true);
```

```
xhr.responseType = "json";
```

```
xhr.setRequestHeader("Content-Type", "application/json");
```

```
xhr.send(JSON.stringify(data));
```

```
xhr.onload = function (e) {
```

```
    if (xhr.readyState === xhr.DONE) {
```

```
        if (xhr.status !== 201) {
```

```
            helper.basicMsgDialog('班级创建失败，重新试一次吧');
```

```
            self.disabled = false;
```

```
            return;
```

```
        }
```

```
        helper.basicMsgDialog('班级创建成功，去加入班级吧');
```

```
        self.disabled = false;
```

```
    }
```

```
}
```



```

xhr.onerror = function (e) {
    helper.basicMsgDialog('抱歉，班级创建失败了，检查一下网络吧');
    self.disabled = false;
}

```

本应用中采用了第二种方式, 可以看到, 需要传入三个参数, 第一个参数是请求方法, 第二个参数是请求 URI, 第三个参数是否异步执行。

6.2.4 页面控件的实现

使用 WinJS 开发 UWP 应用, 应用在生成以及页面跳转的时候, 会将即将要展示的页面<body>部分嵌入到主页面, 页面控件的实现如下:

HTML 引入页面控件:

```

<div class="section" data-win-control="WinJS.UI.PivotItem"
data-win-options="{ isHeaderStatic: true, header: '通知' }">
    <div data-win-control="PageControls.notice"></div>
</div>

```

这里采用了 Pivot(即透视导航布局), 页面控件关键应用:

```
<div data-win-control="PageControls.notice"></div>
```

JS:

```

(function () {
    "use strict";

    var noticePage = WinJS.UI.Pages.define("/pages/notice/notice.html", {
        // 每当用户导航至此页面时都要调用此功能。它
        // 使用应用程序的数据填充页面元素。
        ready: function (element, options) {
            // TODO: 在此处初始化页面。
        },
        unload: function () {
            // TODO: 响应导航到其他页。
        },
        updateLayout: function (element) {
            /// <param name="element" domElement="true" />

            // TODO: 响应布局的更改。
        }
    });
}

```

```

    }
  });
  WinJS.Namespace.define ('PageControls', {
    notice: noticePage
  })
});

```

当进入 notice.html 页面时，应用会初始化该页面，自动执行 ready 中的内容。至此，便实现了页面控件，这是整个应用的骨架。

6.2.5 页面实现效果

在 6.2.3 和 6.2.5 已经将页面实现需要用到的页面控件实现方法以及数据通信采用的方式列出，因此，在这一部分，将直接展示页面实现之后的效果图。

1. 主页面实现效果

打开应用，首先进入主页，在未登录的情况下，没有内容，主页如图 12。

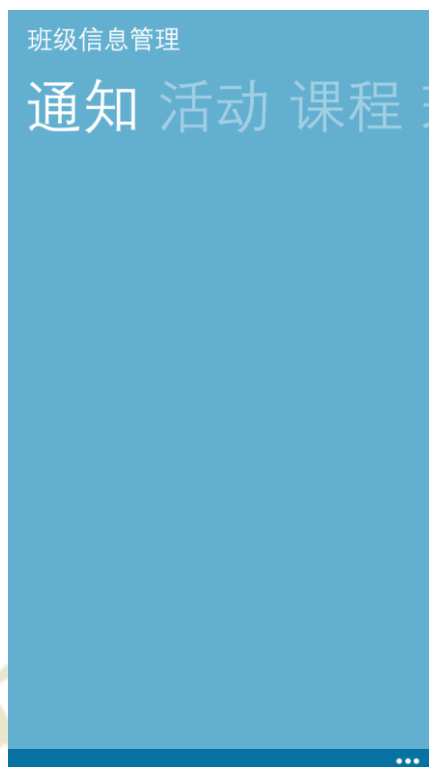


图 12. 未登录主页面

点击右下角的省略号，打开功能菜单，如图 13。

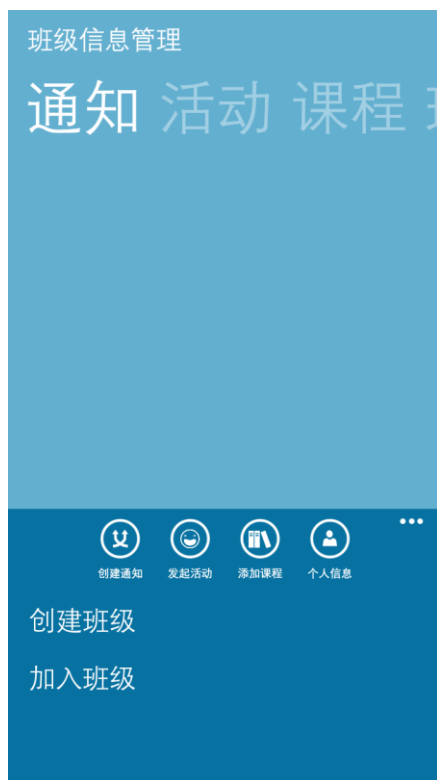


图 13. 功能菜单

2. 登录页实现效果

点击个人信息按钮，进入登录页面，如图 14。



图 14. 登录页

3. 通知页面实现效果

登录之后，返回首页，即通知页面，页面展示通知信息，如图 15。

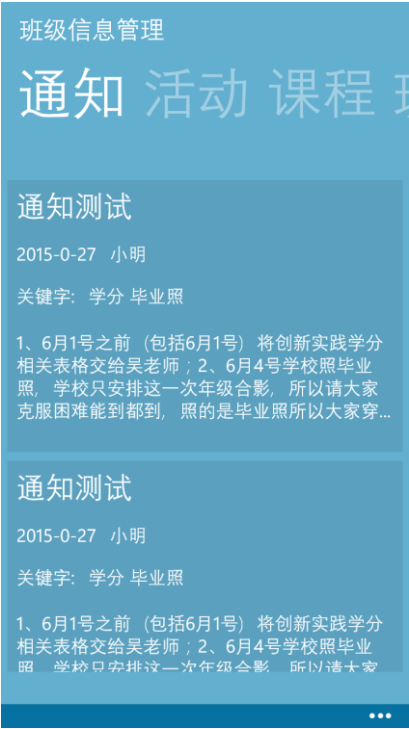


图 15. 通知页面

4. 活动页面实现效果

通过手指在屏幕左右滑动来展示不同栏目下的信息，当向左滑动，展示活动页面，如图 16。

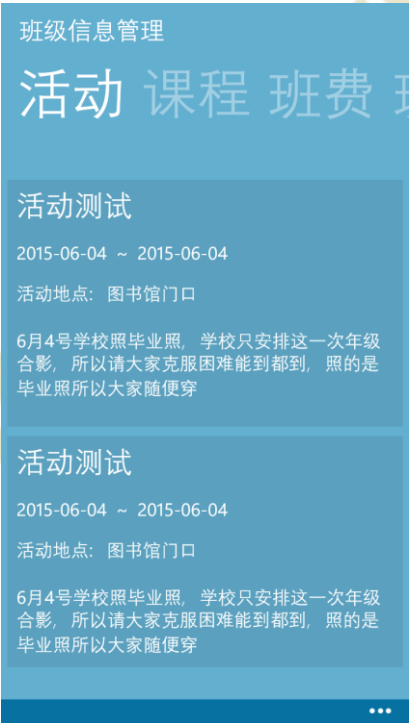


图 16. 活动页面

5. 课程页面实现效果

继续向左滑动，将会看到课程相关的信息，如图 17。

班级管理

课程 班费 班级管理

高数	必修	基础课
不晓得	12345678900	teacher@edu.cn
3 ~ 16周	周3早上5、6节	6601

高数	必修	基础课
不晓得	12345678900	teacher@edu.cn
3 ~ 16周	周3早上5、6节	6601

高数	必修	基础课
不晓得	12345678900	teacher@edu.cn
3 ~ 16周	周3早上5、6节	6601

图 17. 课程页面

6. 班费页面实现效果

继续向左滑动，可以查看班费相关的信息，如图 18。

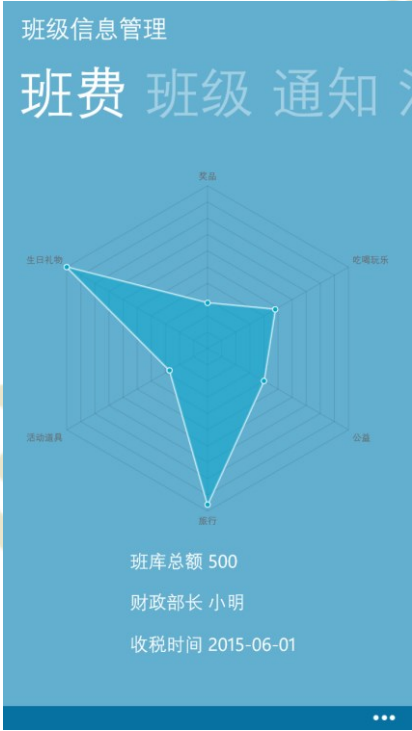


图 18. 班费页面

7. 班级详情页实现效果

继续向左滑动，进入最后一个页面：班级详情页面。如果此后再向左滑动，

将回到通知页面，这是循环的。如图 19。



图 19. 班级详情页

8. 学生名单页面实现效果

在班级页面，点击学生名单按钮，可以查看学生名单及详细信息，如图 20。



图 20. 学生名单页

9. 创建班级页面实现效果

点击右下角省略号，在弹出的菜单中点击“创建班级”，将会进入创建班级

页面，该功能不需要登录。如图 21。

创建班级

必填项

学院

请选择学院

年级

请选择年级

班级

请选择班级

专业

请选择专业

辅导员

请选择辅导员

班长

以下为非必填项

副班长

团委

生活委员

图 21. 创建班级

10. 加入班级实现效果

在弹出的菜单中点击“加入班级”，进入加入班级页面，该功能不需要登录
如图 22。

加入班级

必填项

学号

姓名

班级角色

请选择角色

班级

请选择班级

以下为非必填项

性别

请选择性别

电话

邮箱

宿舍楼

寝室号

图 22. 加入班级

11. 创建通知页面实现效果

在弹出菜单中点击创建通知按钮，可以创建通知，该功能需要登录并且具有班委及以上权限。如图 23。



创建通知

标题

关键字

多个关键字请用空格隔开

内容

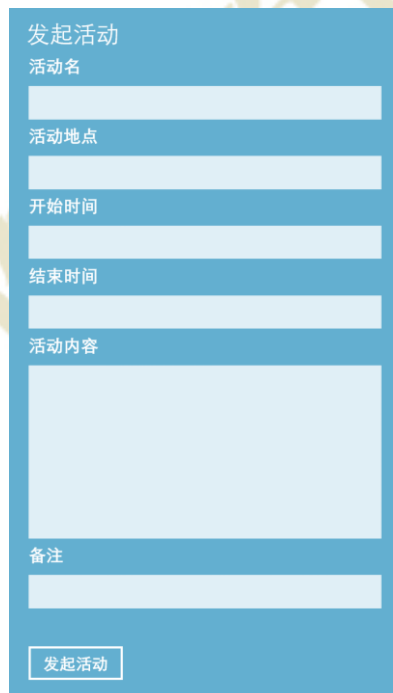
备注

创建通知

图 23. 创建通知

12. 发起活动页面实现效果

在弹出的菜单中点击发起活动按钮，可以发起活动那个，该功能需要登录并具备班委及以上权限。如图 24。



发起活动

活动名

活动地点

开始时间

结束时间

活动内容

备注

发起活动

图 24. 发起活动

13. 添加课程页面实现效果

在弹出菜单中点击添加课程按钮，可以添加课程，如图 25。

添加课程

必填项

课程名

开始周

结束周

上课时间

上课教室

老师

电话

是否必修

选修

课程类型

基础课

图 25. 添加课程

14. 个人信息页面实现效果

在弹出的菜单中点击个人信息按钮，可以进入个人信息页面，查看并且修改个人信息。如图 26

个人信息	
学号	2011124039
新密码	
姓名	小明
电话	12345678900
邮箱	student@live.vn
宿舍楼	15
寝室号	2017
备注	测试数据
<input type="button" value="修改"/>	

图 26. 个人信息

7 调试与测试

7.1 运行环境

Nokia 830 手机：运行 Windows Phone 8.1 OS

7.2 测试原则

- 系统 UI 是否与设计向匹配
- 功能是否完整
- 数据是否正确

7.3 测试内容

7.3.1 网络检测测试

当检测不到网络或者网络无法联网时，应提示用户。如图 27。

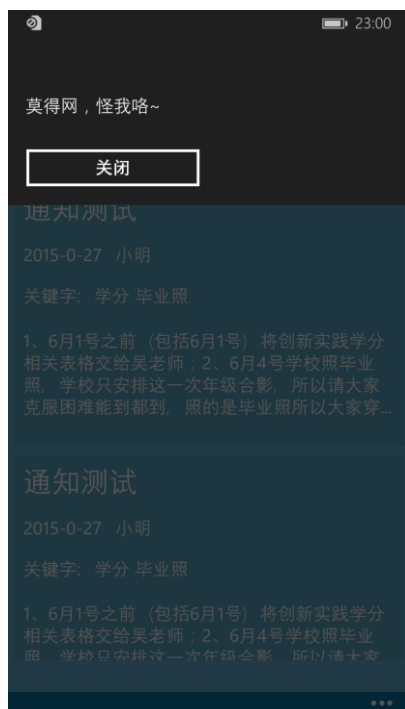


图 27. 无网络提示

7.3.2 登录提示测试

打开应用，若未登录，应提示登录，如图 28。



图 28. 登录提示

7.3.3 输入判空测试

在输入信息时，应当对必填项进行判空，如图 29。



图 29. 判空提示

7.3.4 数据通信测试

GET 请求正常，能请求到需要的数据；

POST 请求失败，未能实现数据的插入；

DELETE 请求失败，未能实现删除数据功能；

PUT 请求失败，未能实现数据的更新。

原因在查找中。

7.4 常见问题和解决方法

- POST/DELETE/PUT 请求失败（严重）；

解决办法：暂无

- 当有多个提示消息同时弹出的时候，会阻塞，并且报错；

解决办法：同样的消息只保留一个，比如提示登录，没有网络。

- 应用栏菜单不能显示

解决办法：退出应用，清空后台，重新打开应用

8 改进建议和措施

1. UI 可以更加流畅，不应该有延迟感；
2. 数据传输应该加密；
3. 表设计有一些问题，需要改进；
4. 操作逻辑不完全符合设计；
5. 服务不完善，只能部分工作。

结 语

此次毕业设计，我尝试着在微软的最新平台 Windows 10 以及 Windows Phone8.1 上进行了第一次 Universal Windows App 的开发。本次开发客户端基于微软的 WinJS 框架，采用 JavaScript + HTML5 + CSS3 技术，服务端采用 Web API 创建符合 RESTful 风格的服务。由于目前 Windows 平台的不成熟，以及国内 WP 的开发者极少，因此，在开发的过程中，遇到了极大的困难。万幸，经过这么长时间的不断努力，终于还是完成了一个不算完整的班级信息管理系统，以及不稳定的服务端。该系统运行较为流畅，能够实现基本的信息查询，添加以及数据的图形化显示。不过，还是有很多做的不好的地方。首先，UI 的响应有时会有迟滞，有时候会闪退，当提示信息过多，会出现阻塞以致应用崩溃，最重要的是，服务端不能够正常的提供服务，使得本次毕业设计不尽完整。因此，在以后的学习中，需要更多的了解服务端的处理方式，以及前后端的交互。

最后，我要特别的感谢昌燕老师和牟映洲同学，感谢昌燕老师在百忙之中抽出时间指导我的毕业设计以及论文，感谢牟映洲同学无私的为我修改论文，因为他们的帮助，我才如期的完成了毕业设计。

参考文献

- [1] Peter, Brian Albers, Frank Salim. Pro HTML5 Programming Second Edition[M].
北京: 人民邮电出版社, 2012. 10
- [2] Microsoft Windows 开发人员中心. 快速入门: 添加 WinJS 控件和样式 (HTML) [EB/OL].
Microsoft Windows 开发人员中心
- [3] Microsoft Windows 开发人员中心. 添加消息对话框 (HTML) [EB/OL]. Microsoft
Windows 开发人员中心
- [4] Microsoft Windows 开发人员中心. 添加应用栏和工具栏 (HTML) [EB/OL]. Microsoft
Windows 开发人员中心
- [5] Microsoft Windows 开发人员中心. 快速入门: 添加 Page 控件 (HTML) [EB/OL].
Microsoft Windows 开发人员中心
- [6] Microsoft Windows 开发人员中心. Windows 运行时应用的 Windows API 参考 [EB/OL].
Microsoft Windows 开发人员中心
- [7] 苏勇. 深入理解 HTTP 协议 [EB/OL]. 360doc 个人图书馆

致 谢

本文是在吕燕老师的热情关心和指导下完成的，他渊博的知识和严谨的治学态度使我受益匪浅，对顺利完成本课题起到了极大的作用。在此向他表示我最衷心的感谢！

在论文完成过程中，本人还得到了牟映洲同学的热心帮助，本人向他们表示深深的谢意！

最后向在百忙之中评审本文的各位专家、老师表示衷心的感谢！

作者简介：信息安全工程学院 物联网工程 111 班 初级前端工程师

姓 名： 万俊嘉

性别： 男

出生年月： 1991/07/23

民族： 汉

E-mail: new_start@live.cn

声 明

本论文的工作是 2015 年 01 月至 2015 年 06 月在成都信息工程学院系完成的。文中除了特别加以标注地方外，不包含他人已经发表或撰写过的研究成果，也不包含为获得成都信息工程学院或其他教学机构的学位或证书而使用过的材料。

关于学位论文使用权和研究成果知识产权的说明：

本人完全了解成都信息工程学院有关保管使用学位论文的规定，其中包括：

(1) 学校有权保管并向有关部门递交学位论文的原件与复印件。

(2) 学校可以采用影印、缩印或其他复制方式保存学位论文。

(3) 学校可以学术交流为目的复制、赠送和交换学位论文。

(4) 学校可允许学位论文被查阅或借阅。

(5) 学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

除非另有科研合同和其他法律文书的制约，本论文的科研成果属于成都信息工程学院。

特此声明！

作者签名：万俊嘉
2015 年 05 月 21 日