
基于微信小程序的教学管理平台的设计与实现

摘要

微信小程序是一款不用下载就可以使用软件。也体现了“用完即走”的理念，应用将无处不在随时可用。用户通过扫描二维码就可以打开应用，使用便捷。

现有的教学平台一般都要开发几款不同平台的软件。本课题的目的旨在借助微信小程序开发出一款不需要下载安装也能实现教学管理目的的软件。技术上采用微信小程序特有的原生态的语言，它相当于把 CSS、HTML 和 JavaScript 融合在一起。数据库方面采用阿里云的云端 MYSQL 数据库。整个系统业务逻辑简单，界面风格简约。功能方面本着方便师生处理正常课堂内外的事物的原则；学生可以完成签到，成绩查看，以及查看老师发布的作业。老师可以发布作业，查看所有人的成绩。老师和学生都可以在讨论区讨论问题。

关键词：小程序 便捷 查询 发布 教学平台

Design and Implementation of a Teaching Management Platform Based on WeChat Applet

Abstract

WeChat applet is a software that can be used without download. It also embodies the concept of "run out and go", and applications will be available everywhere at any time. The user can open the application by scanning the QR code.

Now,existing teaching platforms generally have to develop softwares for different platforms. The purpose of this project is to use the WeChat applet to develop a software that can achieve teaching management without downloading and installing. It uses the original ecological language unique to the WeChat applet. It is equivalent to integrating CSS, HTML, and JavaScript. Database uses Alibaba Cloud's cloud MYSQL database.The business logic of the entire system is simple and the interface style is simple. Functionality is based on the principle of facilitating teachers and students to handle things in and outside the normal classroom; students can complete check-in, check scores, and view assignments posted by teachers. Teachers can post assignments to see everyone's grades. Teachers and students can discuss issues in the discussion area.

Key words:Small program;Convenient;Inquire;release;teaching platform

目录

1 绪论	1
1.1 课题研究背景.....	1
1.2 课题研究意义.....	1
1.3 课题研究内容和成果.....	1
2 相关工作	2
2.1 小程序开发工具.....	2
2.2 开发核心.....	2
2.2.1 框架主体文件.....	2
2.2.2 框架页面文件.....	2
2.3 布局.....	2
2.4 组件.....	2
2.5 API.....	3
2.5.1 网络.....	3
2.5.2 媒体.....	3
2.5.3 文件.....	3
2.5.4 数据缓存.....	3
2.5.5 位置.....	4
2.5.6 设备.....	4

2.5.7 界面.....	4
2.5.8 开放接口.....	4
3 可行性分析.....	5
3.1 技术可行性.....	5
3.2 操作可行性.....	5
3.3 经济可行性.....	5
4 需求分析.....	6
4.1 运行环境需求分析.....	6
4.2 系统功能需求分析.....	6
4.3 系统流程设计.....	6
4.3.1 系统开发流程.....	6
4.3.2 系统操作流程.....	7
4.4 系统用例图分析.....	8
5 系统分析.....	13
5.1 系统交互过程分析.....	13
5.2 整理分析类.....	16
6. 系统设计.....	18
6.1 系统概要设计.....	18
6.1.1 系统的总体设计.....	18
6.1.2 系统的设计原则.....	18

6.2 数据库设计.....	18
6.2.1 E-R 图设计	19
6.2.2 数据库表结构设计.....	21
6.2.3 数据库连接	22
6.3 bmob 的使用	22
7 系统具体实现	25
7.1 学生模块的实现.....	25
7.1.1 学生模块的简述.....	25
7.1.2 签到模块实现.....	25
7.1.3 成绩查询的实现.....	27
7.1.4 作业查看的实现.....	27
7.1.5 交流区的实现.....	29
7.2 老师模块的实现.....	29
7.2.1 老师模块的简述.....	29
7.2.2 发布作业模块的实现.....	29
8 系统测试	31
8.1 系统测试的意义	31
8.2 具体测试任务.....	32
8.2.1 界面的测试.....	32
8.2.2 作业发布测试.....	33

8.2.3 成绩查询测试.....	33
8.2.4 签到系统的测试.....	34
8.2.5 讨论区的测试.....	35
8.3 测试分析	35
结论.....	36
参考文献.....	37
致谢.....	38

1 绪论

1.1 课题研究背景

微信小程序是一款不用下载就可以使用的应用程序，它实现了应用“触手可及”的梦想，扫一扫或者搜一下即可打开应用，并且即用即走。它通过极低的开发成本，大量简化了相关的开发操作和应用信息获取，结合 HTML5 和微信 API，几乎能开发出类似 APP 的大多数功能。

1.2 课题研究意义

微信是目前使用人数最多，应用最广泛的一款即时通讯应用软件。基于微信小程序的教学管理平台，可以更好完成教学任务，提高教学效率。学生可以完成课前签到，随时随地完成提交作业以及请假等事物。老师可以随时随地上传批改作业。极大的丰富了教师和学生的日常课堂交流。

1.3 课题研究内容和成果

本课题借助微信小程序开发出一款教学管理平台。基于微信小程序的教学管理平台，可以更好完成教学任务，提高教学效率。学生可以完成课堂签到的功能，取代了传统的班长或者老师点名签到的方式。学生能够在列表里查看到老师发布的作业。在学习讨论区，学生和老师可以进行互动交流，不仅讨论学习方面，也可以讨论生活方面等。另一个比较重要的功能就是可以查看成绩。老师负责把成绩录入教务系统，和发布作业。同时可以和学生在讨论区聊天。

2 相关工作

2.1 小程序开发工具

为了帮助开发者开发和调试微信小程序，微信推出了微信开发者工具，既可以完成公众号的调试以及小程序的开发与调试两种功能。使用微信 Web 开发者工具可以完成对小程序的开发调试、代码的查看与编辑以及预览和发布等功能。

目前微信 Web 开发者工具支持 Windows 以及 Mac 平台。它和其他开发工具不同的是需要先扫码登录才能使用。另一个特色就是，在创建新的项目时候，需要添加 AppID，否则部分功能如预览不能使用。

2.2 开发核心

2.2.1 框架主体文件

框架主体文件由 `app.json`、`app.js`、`app.wxss` 构成，这 3 个文件必须放置在项目根目录，其中 `app.js` 是整个文件的入口文件，控制整体逻辑。`app.wxss` 是小程序的公共样式，对于整个文件夹都是生效的。但是它不是必须存在的。`app.js` 和 `app.json` 是必须存在的。

2.2.2 框架页面文件

框架控制了整个小程序的页面路由，并且提供以页面完整的生命周期。开发者需要做的只是将页面的数据，方法和生命周期函数注册进框架中，其他操作由框架处理，开发者不需要关心其他问题。

2.3 布局

微信小程序页面布局方式采用 Flex 布局格式，它是 W3c 在 2009 年提出的一种新的网页排版方案，可以简便、完整地实现各种响应式页面布局。Flex 布局的主要特点是：通过设定容器与子元素的规则，使所有 view 组件在主轴与侧轴上合理地自动分配。同时可以实现在水平和一个方向上的伸缩，样式层可以调换或者重排顺序。

2.4 组件

框架为开发者提供了一系列基础组件，开发者可以通过选择这些基础组件进行快速开发。组件是视图层的基本组成单元。组件自带一些功能与微信风格的样式。一个组件通常包括开始标签和结束标签，属性用来修饰这个组件，内容在两个标签之内。常用的组件有：视图组件、表单组件、基础内容、导航、媒体组件、地图和画布等。

2.5 API

小程序框架提供丰富的微信原生 API，可以调起微信提供的能力，如获取用户信息、本地存储、支付功能等。目前提供的 API 可分为八大类：网络 API、多媒体 API、文件 API、数据存储 API、位置 API、设备 API、界面 API 以及开放接口。

2.5.1 网络

每个微信小程序需要事先设置一个通讯域名，小程序可以跟指定的域名与进行网络通信。包括普通 HTTPS 请求（request）、上传文件（uploadFile）、下载文件（downloadFile）和 WebSocket 通信（connectSocket）。服务器域名请在 小程序后台-设置-开发设置-服务器域名 中进行配置，配置时需要注意：域名只支持 https（request、uploadFile、downloadFile）和 wss（connectSocket）协议；域名不能使用 IP 地址或 localhost，且不能带端口号；域名必须经过 ICP 备案；出于安全考虑，api.weixin.qq.com 不能被配置为服务器域名，相关 API 也不能在小程序内调用。小程序必须使用 HTTPS 请求。网络请求的 referer header 不可设置。其格式固定为 https://servicewechat.com/{appid}/{version}/page-frame.html，其中 {appid} 为小程序的 appid，{version} 为小程序的版本号，版本号为 0 表示为开发版。

2.5.2 媒体

小程序可以调用由微信调用的从本地相册选择图片，或者使用相机拍照的 API, 还可以操作网络上的图片、音频、文件，甚至对视频进行相关的播放和暂停关闭等功能。

2.5.3 文件

使用 API 对小程序中出现的各类文件进行操作。例如常用的有 `wx.saveFile` 即保存本地文件，`wx.getSavedFileList` 即获取本地已保存的文件列表等。

2.5.4 数据缓存

每一个微信小程序都可以有自己的本地缓存，可以通过 `wx.setStorage(wx.setStorageSync)`、`wx.getStorage(wx.getStorageSync)`、`wx.clearStorage(wx.clearStorageSync)` 对本地缓存进行设置、获取和清理。本地缓存最大为 10MB。LocalStorage 是永久的本地存储，但是为了防止用户在更换设备后丢失信息，所以对于重要信息还是建议存储在服务器端，通过请求获取。数据的存储主要通过“键值对”的形式实现。

2.5.5 位置

包括获取位置 (`wx.getLocation`)，查看位置 (`wx.openLocation`) 以及地图组件控制。

2.5.6 设备

微信端的权限可以通过提供的相关 API 实现获取系统信息、设备信息 (如网络状态、重力感应、罗盘、拨打电话和扫码等)。

2.5.7 界面

在创建微信小程序界面时，不仅需要使用微信小程序的组件，还要通过调用其相关的 API 来实现组建的使用或完成组件之间的逻辑关系。

2.5.8 开放接口

除以上几种之外，微信官方还提供了一些开放接口，如果你接触过微信公众号的开发，就会发现这些接口大部分实现起来差不多。主要包含登录、获取用户信息、使用微信支付、模版消息、客服消息、分享以及获取二维码等几类功能。

3 可行性分析

3.1 技术可行性

本文所设计的微信小程序使用的 API 接口均来自微信官方所提供。基于微信小程序的教学管理平台所采用的技术、组件和 API 等均可以在现有的资源上实施。WXSS 是微信小程序的 CSS，它对 CSS 的大部分标准可以支持，只是在此基础上稍作扩展，因此我们完全可以直接使用 CSS 语法编写 WXSS 语言。

WXML 是一套类似于 HTML 的标签语言。语法与 HTML 总体上一致，只是稍有改动，几乎可以忽略不计。

逻辑层，其主要技术手段是 JavaScript，用于一些数据的定义和 API 方法的实现。

综上，虽然微信小程序开发的技术不是原生态的 JavaScript、HTML 和 CSS。但若是把上述三种技术封装起来的一门技术。在技术上完全可行。

3.2 操作可行性

本文所设计的微信小程序使用微信官方提供的微信 Web 开发者工具。所要求的软硬件都在可操作空间内，完全可以达到预期的效果。

3.3 经济可行性

由于微信小程序越来越开放，对开发者的要求也在慢慢降低，基本上不需要花费多少财力和物力就可以实现。

4 需求分析

4.1 运行环境需求分析

微信 Web 开发者工具 v1.01.1711300

Mysql 版本: Mysql 5.0

操作系统版本: Windows 10

4.2 系统功能需求分析

需求分析是整个软件开发的基础。系统需求分析的过程中，一般可以分为如下几个步骤：第一步需要先确认目标了解范围，第二步调查需求，第三步确认系统内容，然后编写需求文档。完善需求分析可以让模块设计以及编码更加简洁明了。

对于基于微信小程序的教学管理平台的需求分析，主要分为一下两个部分。

(1) 事务管理，负责师生课堂沟通工作；

(2) 成绩管理，负责学生成绩管理。

在事务管理方面：

签到系统

老师根据学生签到记录，统计学生的出勤情况。

(2) 布置作业

老师可以添加作业

在成绩管理方面：

(1) 成绩录入：

录入学生考试成绩

(2) 成绩查询:

教师可以查询所有学生考试成绩，一个学生只能查看自己的成绩，不允许查询别人的考试成绩。

4.3 系统流程设计

4.3.1 系统开发流程

教学管理平台开发时，首先需要进行需求分析，然后对系统进行模块的划分，功能的设计，数据库设计，然后对各模块详细设计和代码编写，之后对系统功能模块进行整合，最后进行测试和调试，流程如图 3-1 所示。

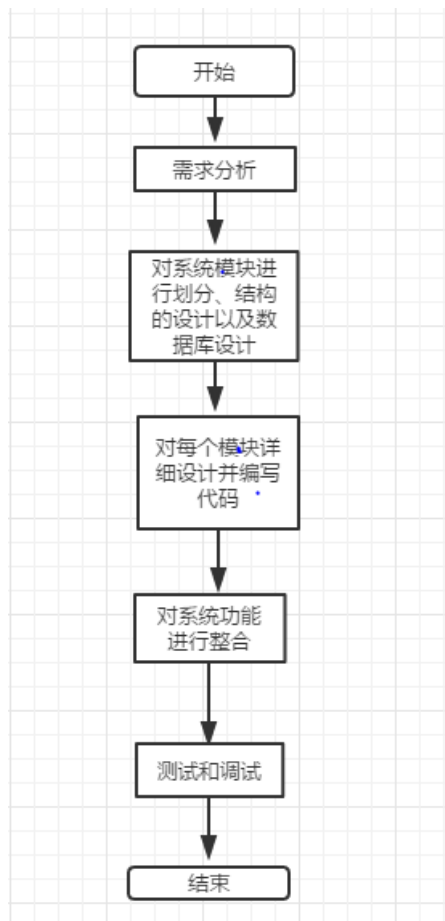


图 4-1 系统开发流程图

4.3.2 系统操作流程

用户打开微信首先展现的是首界面，用户可以点击签到。在作业添加上，老师可以发布作业，然后弹出来一个对话框完成作业的添加。课堂讨论环节，学生和老师可以进行聊天，交流处理一些事物。还可以完成成绩查询的操作。流程图如图 3-2 所示。

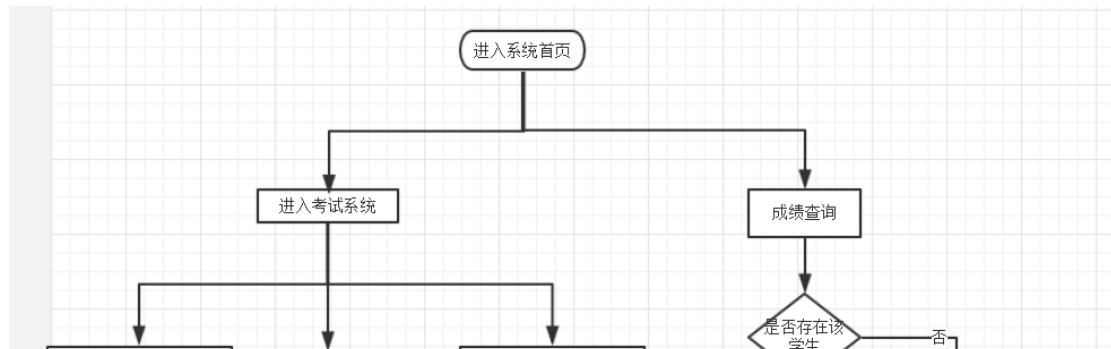


图 4-2 系统操作流程

4.4 系统用例图分析

系统中的核心是学生和老师。

(1) 学生用户：学生具备课堂签到的功能。还可以查看老师发布的作业，同时可以查看老师发布的成绩，学生也可以加入课堂讨论。具体用例图如 3-3 所示

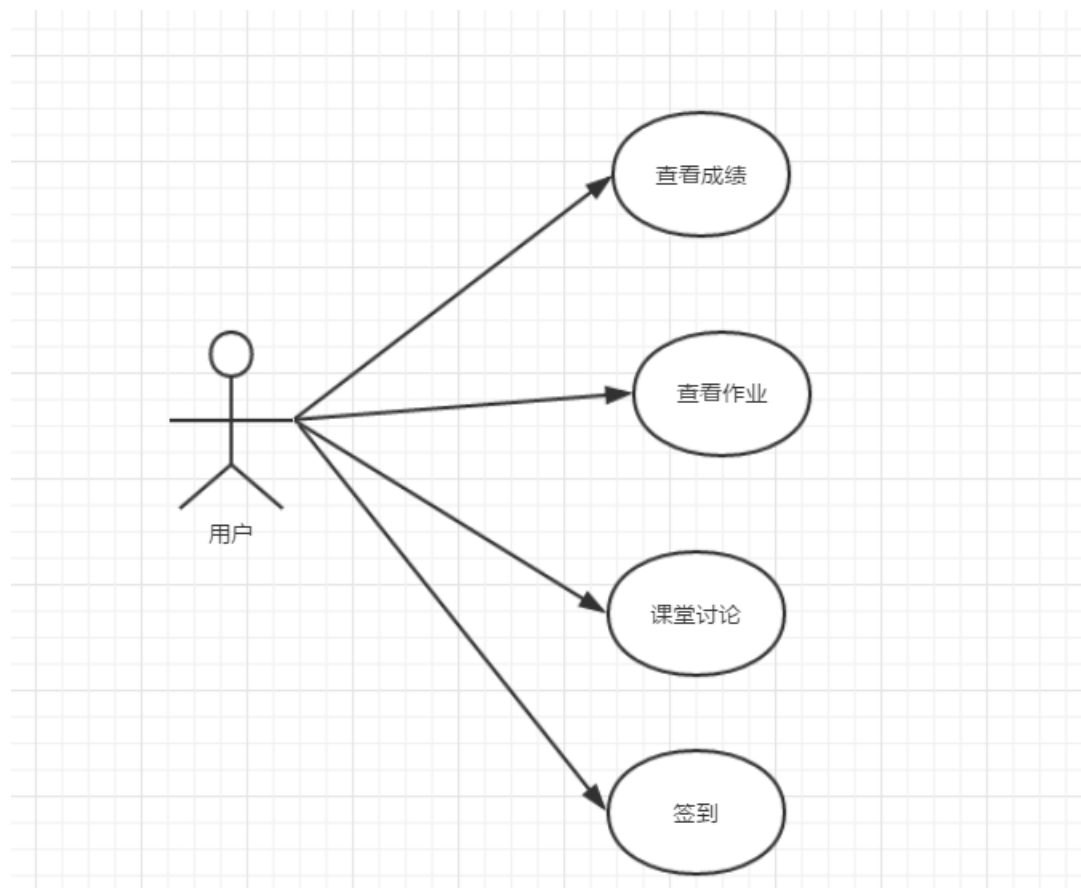


图 4-3 学生用例图

查看成绩				

表 4-1 学生用户用例图

老师用户：老师可以把成绩录入系统中，同时可以查看所有学生的成绩。老师同时具备添加作业和课堂答疑功能。老师也可以参与课堂讨论。如用例图 3-4 所示

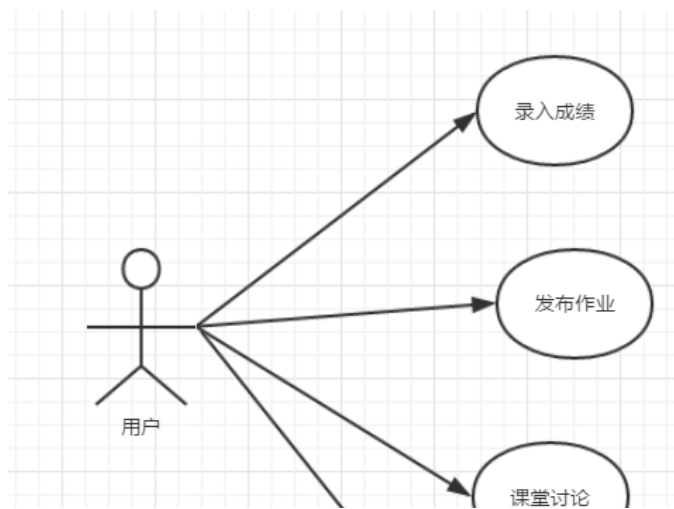


图 4-4 老师用户用例图

会东作业				

课堂答疑	老师	解决学生遗留的问题	进入讨论区	<ol style="list-style-type: none">1. 进入讨论区2. 在文本编辑区提交问题, 提交到服务器在显示区显示
------	----	-----------	-------	--

表 4-2 老师用户用例图

5 系统分析

5.1 系统交互过程分析

序列图也称为“顺序图”，是一种对象间的交互图。用于描述执行系统功能的各个不同对象角色之间相互协作的顺序关系、相互传递的消息，显示跨越多个对象的系统控制流程。它强调的是事件发生的时间以及消息传递的先后次序，它和协作图同属于系统动态模型中的重要组成部分。

以下是本课题在作业查询、作业发布、成绩查询、签到和讨论区功能的序列图：

作业查询：

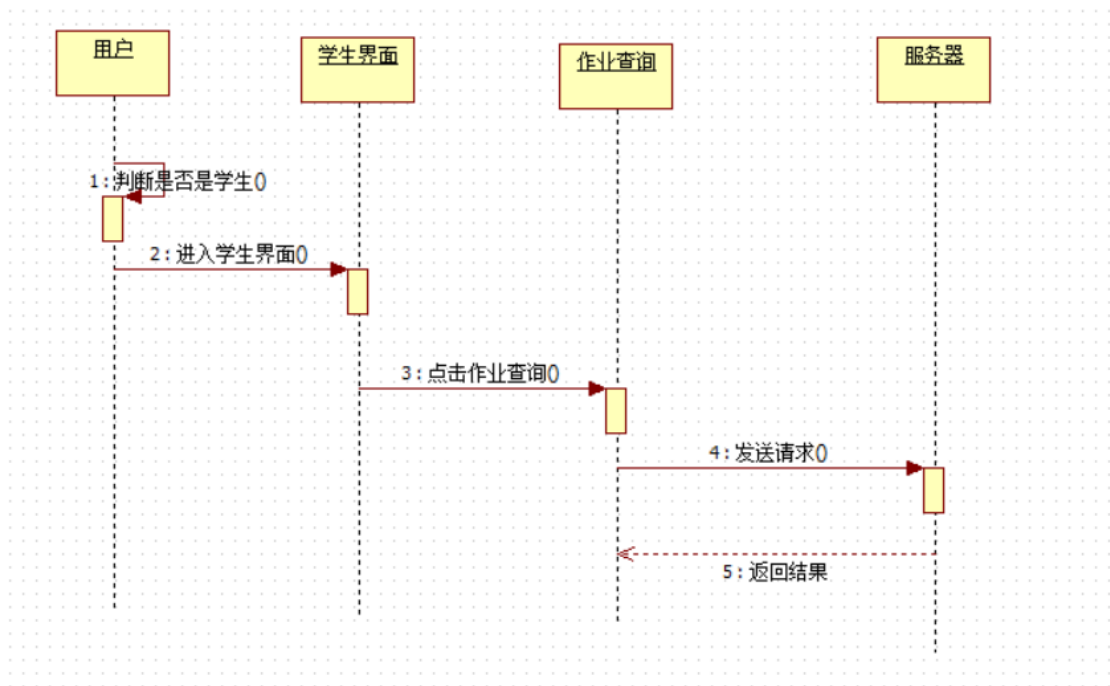


图 5-1 作业查询序列图

作业发布:

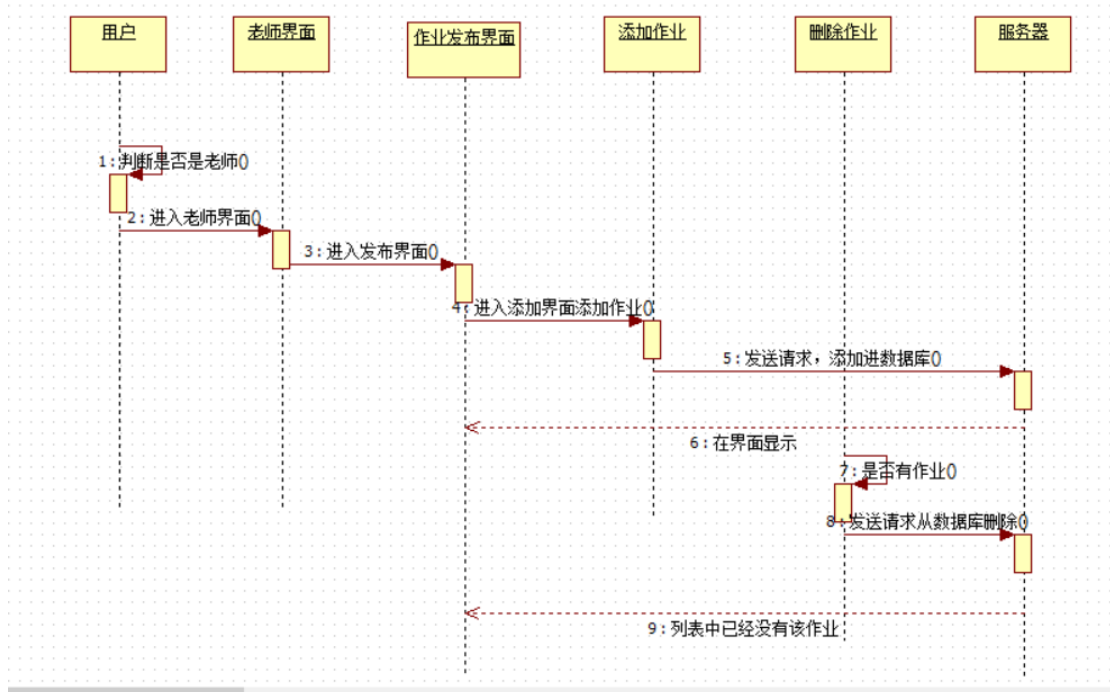


图 5-2 作业发布序列图

成绩查询:

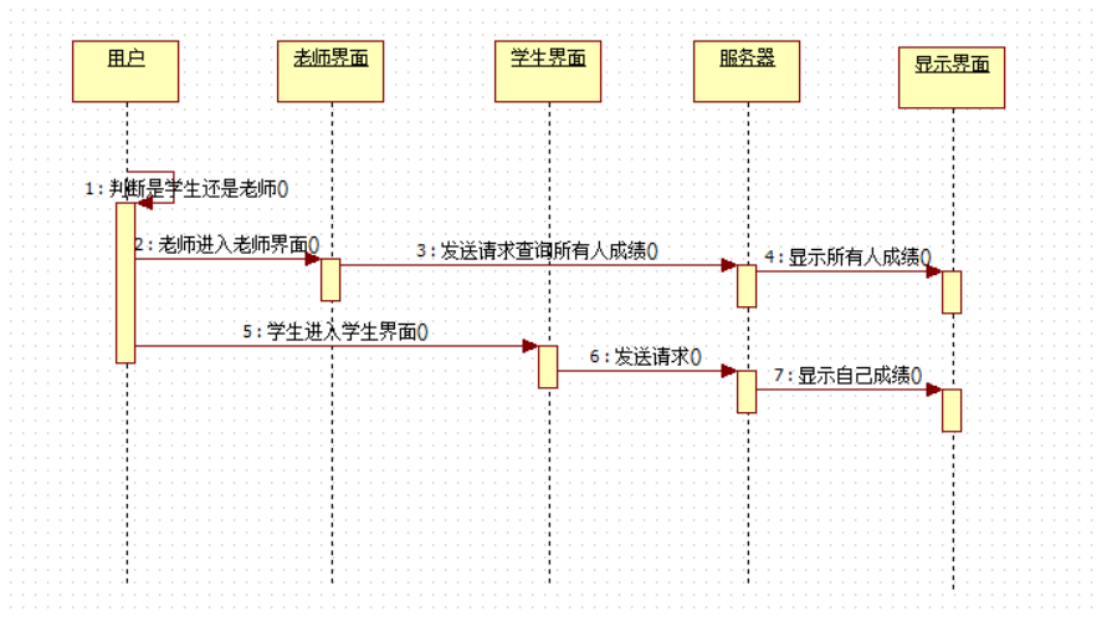


图 5-3 成绩查询序列图

讨论区：

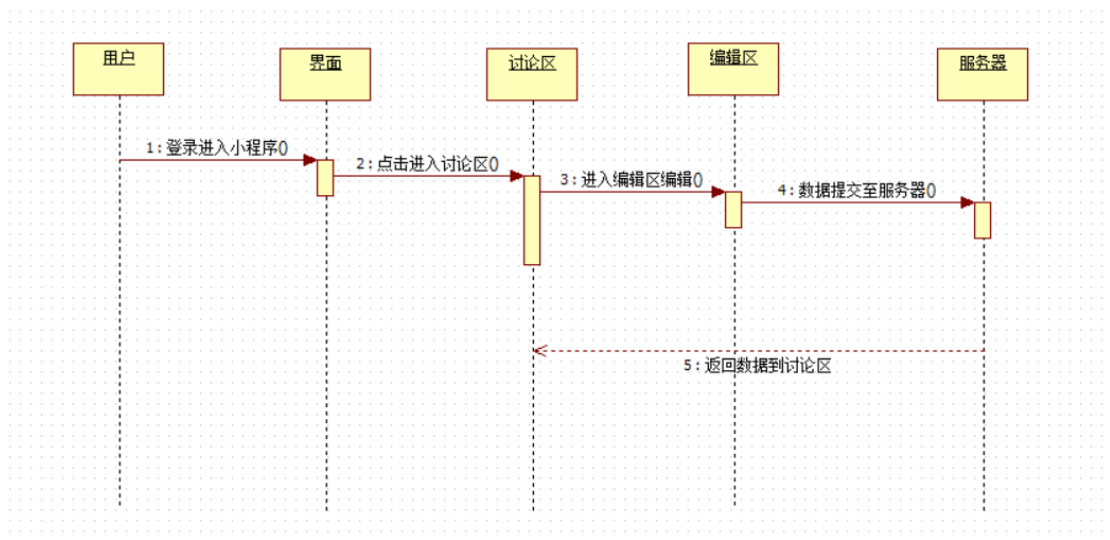


图 5-4 讨论区序列图

签到：

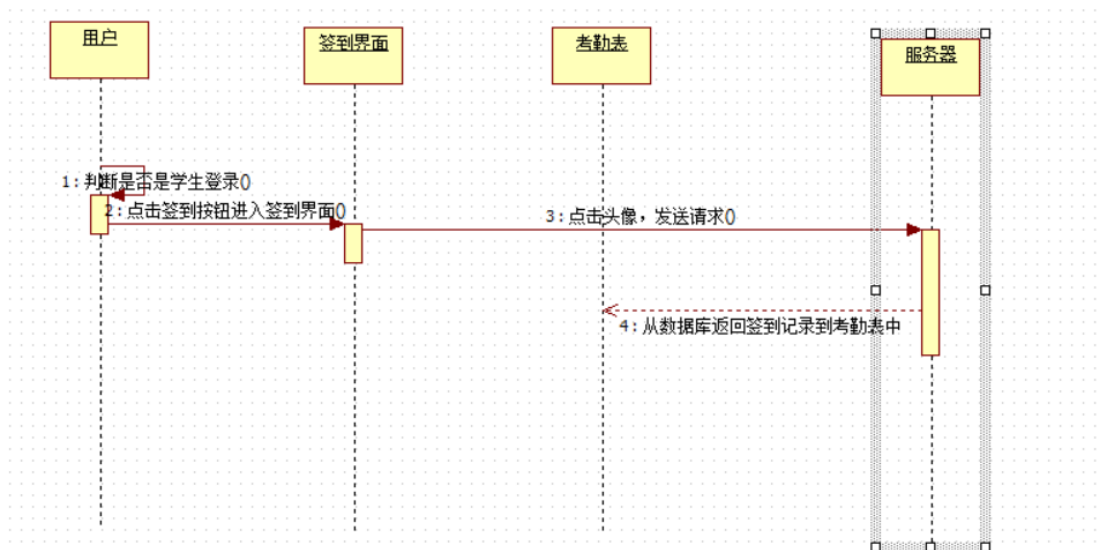


图 5-5 签到序列图

5.2 整理分析类

类图描述了系统的静态结构，它不仅仅定义系统中的类，描述类之间的关系，还包括类的内部结构，在系统的整个生命周期中都是有效的。在系统分析与设计阶段，类通常可以分为实体类、控制类和边界类三类。

实体类：

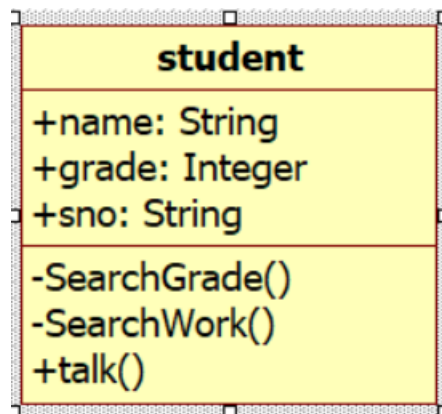


图 5-5 学生实体类

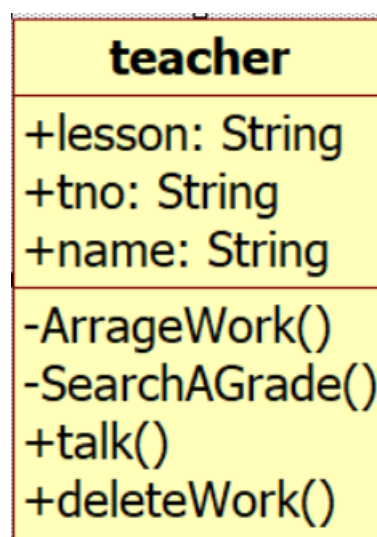


图 5-5 老师实体类

控制类:

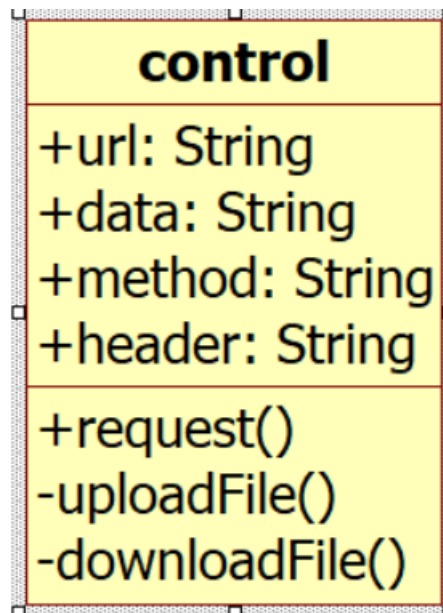


图 5-6 数据操作控制类

6. 系统设计

6.1 系统概要设计

6.1.1 系统的总体设计

基于微信小程序的教学管理平台是一款面向师生课内课外交流的一款应用程序。微信是最近几年来最普及的一款网络通信软件。依托微信积累起来的用户量，利用更新更普及更便捷的微信小程序，进行开发。

在系统设计的过程中，应该完成课堂签到、发布作业、日常交流、查看成绩等功能。根据学生和老师角色的不同。设置相应的功能。学生的核心功能是签到更功能。当然也可以查看自己的成绩。但是只能查看自己的成绩。学生可以在列表查看老师发布的作业。老师可以添加作业，添加完之后可以在列表显示。在交流区，是对老师和学生都是开放的没有权限限制。

6.1.2 系统的设计原则

在整个系统的开发过程，应该本着方便快捷，操作简单，效率高，功能齐全的原则。除了保证上面描述的几个原则之外，还要保证以下几个方面，这样才更加完整。

(1) 实用性原则：确实可以帮助老师和学生在课堂内外的交流，可以用更加先进的技术手段解决一些传统问题。

(2) 简单易用性原则： 不管是从界面的简洁性还是功能上，一定要保证都是需要的功能，逻辑也要保持简单明了，这也是微信小程序的初衷。

(3) 稳定性原则：教学管理平台采用的技术是当前比较新的技术，但是由于该教学管理平台的业务逻辑和功能目前还是比较简单的，现有的技术完全可以满足需要。从系统结构，数据处理等方面做了进一步修改，又保证了系统的稳定性。

(4) 易于维护性原则：在需求分析阶段就采用模块化思想，然后再进行代码的编程，不仅让代码清晰整洁，也方便后期维护，对于版本的升级和功能的进一步扩展都奠定了基础。

(5) 安全性原则 保障访问系统的安全性。由于微信小程序自己的安全性原则，基本层面上可以确保微信小程序的安全。

6.2 数据库设计

6.2.1 E-R 图设计

E-R 图就是实体-联系图。为了描述客观世界，提供了实体类型、联系和属性的方法。它可以描述现实世界的关系。我们用矩形表示实体，用椭圆形表示属性。在教学管理平台中主要有学生和教师实体类，当然还有成绩实体类。以下是学生、成绩和老师实体类图；

(1) 学生实体

学生实体包括微信号，性别，姓名，班级属性。如图 6-1 学生实体图

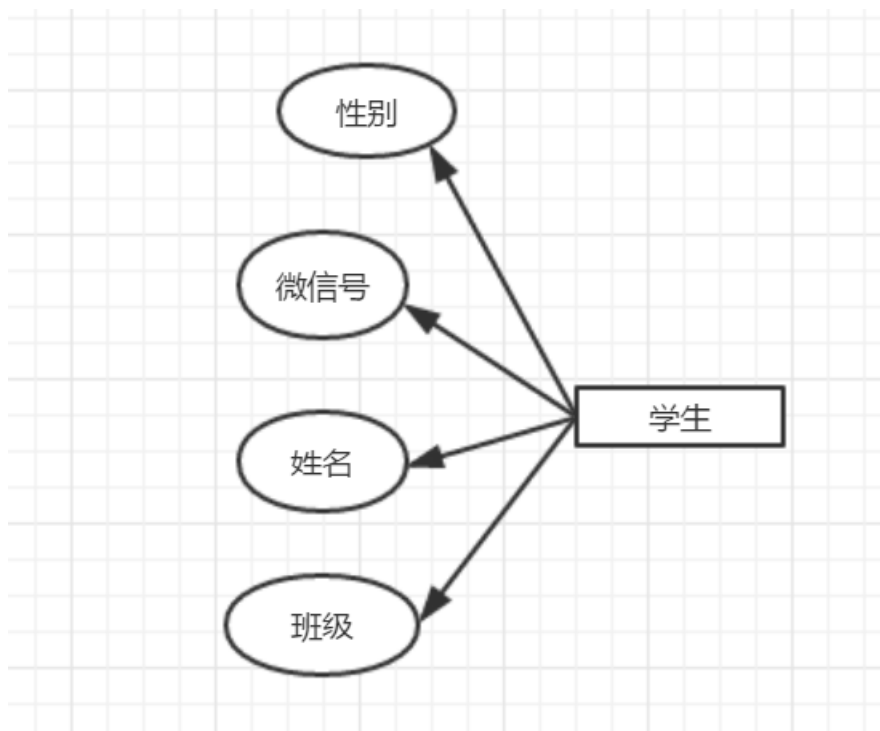


图 6-1 学生实体图

(2) 老师实体

老师属性主要包括微信号，姓名。老师实体图如图 6-2

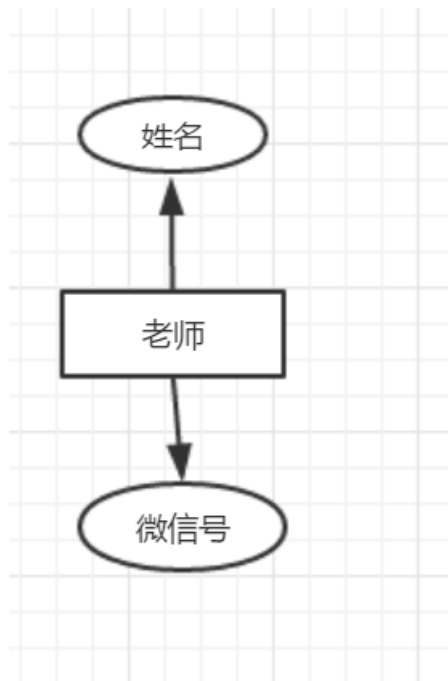


图 6-2 老师实体图

(3) 教学管理平台涉及的主要实体的 E-R 图 如图 6-3 教学管理平台 E-R 图

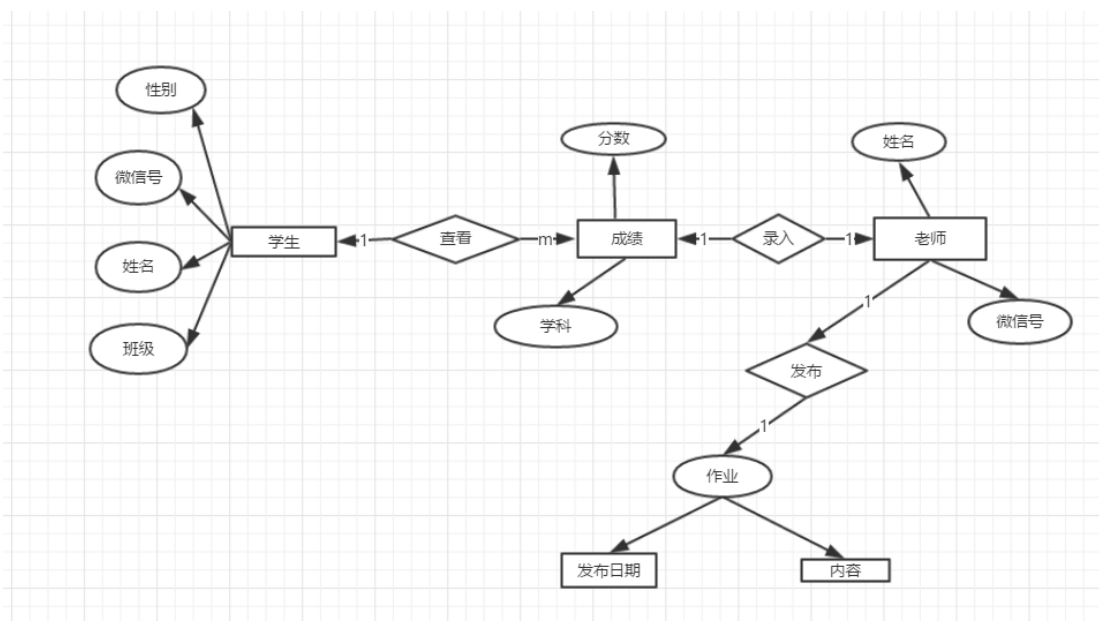


图 6-3 教学管理平台 E-R 图

6.2.2 数据库表结构设计

系统的数据库表是存储和管理数据的最基本单元，该系统的数据库表主要涉及学生表、老师表、成绩表和作业表。上述四个表的结构设计如下所示。

(1) 学生表：主要存储学生的一些个人信息，例如微信号，姓名，班级等如表 6-1 学生表

表 6-1 学生表

字段名	类型	说明
SWechatID	varchar(15)	微信号
SName	varchar(20)	姓名
Sex	varchar(15)	性别
Class	varchar(15)	班级

(2) 老师表：主要存储学生的一些个人信息，例如微信号，姓名，等如表 6-2 老师表

表 6-2 老师表

字段名	类型	说明
TWechatID	varchar(15)	微信号
TName	varchar(20)	姓名

(3) 主要存储有关成绩的的一些信息，例如成绩和学科等如表 6-3 成绩表

表 6-3 成绩表

字段名	类型	说明
Grade	Int(4)	成绩
Lesson	varchar(20)	学科

(4) 主要存储作业的一些信息，例如日期和内容等 如表 6-4 作业表

表 6-4 作业表

字段名	类型	说明
Date	varchar(20)	日期
Things	varchar(500)	内容

6.2.3 数据库连接

微信小程序要连接数据库目前有两种方法可供参考。第一种是通过服务端 web 应用程序，提供 https 接口让小程序调用，这个应用程序可操作本地数据库（这里使用的都是云端数据库，但方法类似）；第二种就是通过微信封装的方法存储数据。

这里采用的是第一种。首先写一个服务端的程序接口，查询数据库，然后指定参数、请求方式等等、小程序直接通过 url，带上一些参数请求服务端的程序，服务端程序再从数据库读取数据，返回到接口，接口再返回到小程序。

6.3 bmob 的使用

在本课题的小程序的设计过程中，使用到了 bmob 技术。Bmob 实际就是后端云，说的通俗一点，他是在云端运行的数据库后台加上服务器后台。它最大的优点就是：我们不必去考虑一些复杂的后台服务器，后台的数据库大件以及数据库的维护，这样我们可以更好的专注于小程序本身的业务。

我们要先注册一个 bmob 帐号，然后登录 bmob 控制台。载在小程序服务器域名界面添加 bmob 的安全域名，下载微信小程序所对应的 SDK，解压之后将其中的 js 文件放在小程序的 utils 文件夹下。如图 6-4bmob 文件：

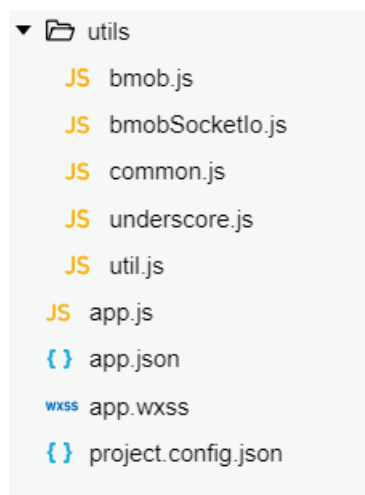


图 6-4bmob 文件

初始化并且引入 bmob。我们首先要在 app.js 中对全局初始化。核心代码如下：

```
var Bmob = require('utils/bmob.js');
Bmob.initialize('XXX','XXXXXXX');
// 注：其中'XXX'为 MyBmobApp 的 Application ID,'XXXXXXX'为其 REST API key
```

再在 bmob 的 js 中引用 bmob，核心代码如下：

```
var Bmob = require('../../utils/bmob.js');
```

接下来我们就可以像操作本地数据库一样操作云端数据库了。API 都封装在 bmob 中。如图 6-5bmob 的部分截图


```

(function (root) {
  var _ = require('underscore.js');
  var Bmob = {};
  Bmob.VERSION = "js0.0.1";
  Bmob._ = _;

  var EmptyConstructor = function () { };

  if (typeof exports !== 'undefined') {
    if (typeof module !== 'undefined' && module.exports) {
      exports = module.exports = Bmob;
    }
    exports.Bmob = Bmob;
  } else {
    root.Bmob = Bmob;
  }

  // Helper function to correctly set up the prototype chain, for subclasses.
  // Similar to `goog.inherits`, but uses a hash of prototype properties and
  // class properties to be extended.
  var inherits = function (parent, protoProps, staticProps) {
    var child;

    _mergeWithPrevious: function (previous) {
      if (!previous) {
        return this;
      } else if (previous instanceof Bmob.Op.Unset) {
        return new Bmob.Op.Set(this.objects());
      } else if (previous instanceof Bmob.Op.Set) {
        return new Bmob.Op.Set(this._estimate(previous.value()));
      } else if (previous instanceof Bmob.Op.AddUnique) {
        return new Bmob.Op.AddUnique(this._estimate(previous.objects()));
      } else {
        throw "Op is invalid after previous op.";
      }
    },

    _estimate: function (oldValue) {
      if (!oldValue) {
        return _.clone(this.objects());
      } else {
        // We can't just take the _.uniq(_.union(...)) of oldValue and
        // this.objects, because the uniqueness may not apply to oldValue
        // (especially if the oldValue was set via .set())
        var newValue = _.clone(oldValue);
        Bmob._arrayEach(this.objects(),
          function (obj) {
            if (obj instanceof Bmob.Object && obj.id) {

```

图 6-5bmob 的部分截图

7 系统具体实现

7.1 学生模块的实现

7.1.1 学生模块的简述

学生模块是教学管理平台的一个主要模块。学生的一个最主要的功能就是签到。学生点击课堂交流可以和老师进行互动，讨论内容，探讨学业。在右下角的成绩查询板块可以查询到自己的成绩。登录界面的核心代码如下：

```
//app.js
var Bmob = require('utils/bmob.js');
Bmob.initialize("e8f1e205232485292f64cb65b3d4f376",
"f9f0d3f1dc689020ffc19fb686da3f44");
App({
  onLaunch: function () {
    // 展示本地存储能力
    var logs = wx.getStorageSync('logs') || []
    logs.unshift(Date.now())
    wx.setStorageSync('logs', logs)
    // 登录
    wx.login({
      success: res => {
        // 发送 res.code 到后台换取 openId, sessionKey, unionId
      }
    })
    // 获取用户信息
    wx.getSetting({
      success: res => {
        if (res.authSetting['scope.userInfo']) {
          // 已经授权，可以直接调用 getUserInfo 获取头像昵称，不会弹框
          wx.getUserInfo({
            success: res => {
              // 可以将 res 发送给后台解码出 unionId
              this.globalData.userInfo = res.userInfo

              // 由于 getUserInfo 是网络请求，可能会在 Page.onLoad 之后才
              // 返回
              // 所以此处加入 callback 以防止这种情况
            if (this.userInfoReadyCallback) {
              this.userInfoReadyCallback(res)
            }
          }
        }
      }
    })
  }
})
```

7.1.2 签到模块实现

学生点击签到按钮进入签到界面，然后点击自己的头像完成签到任务，签到时间会在签到记录中显示出来。核心代码如下：

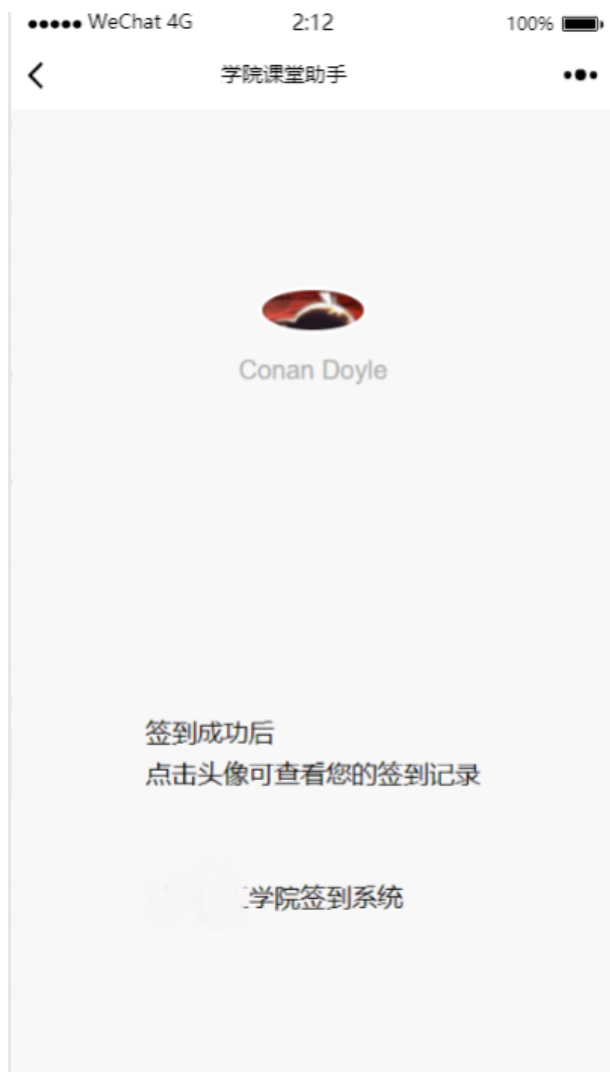


图 7-1 签到界面

```
data: {
  motto: '签到成功后\n 点击头像可查看您的签到记录\n\n\n 学院签到系统',
  userInfo: {},
  hasUserInfo: false,
  canIUse: wx.canIUse('button.open-type.getUserInfo')
},
bindViewTap: function() {
  wx.navigateTo({
    url: '../logs/logs'
  })
},
onLoad: function () {
  if (app.globalData.userInfo) {
    this.setData({
      userInfo: app.globalData.userInfo,
      hasUserInfo: true
    })
  } else if (this.data.canIUse){
    app.userInfoReadyCallback = res => {
      this.setData({
        userInfo: res.userInfo,
        hasUserInfo: true
      })
    }
  } else {
    wx.getUserInfo({
```

7.1.3 成绩查询的实现

在右下角有一个成绩查询，点击之后可以查询到自己的成绩。

7.1.4 作业查看的实现

点击查看按钮即可。核心代码：

```
var objectId = event.target.dataset.id;
wx.showModal({
  title: '操作提示',
  content: '确定要删除? ',
  success: function (res) {
    if (res.confirm) {
      //删除日记
      var Diary = Bmob.Object.extend("dadayu");

      //创建查询对象，入口参数是对象类的实例
      var query = new Bmob.Query(Diary);
      query.get(objectId, {
        success: function (object) {
          // The object was retrieved successfully.
          object.destroy({
            success: function (deleteObject) {
              console.log('删除成功');
              getList(that)
            },
            error: function (object, error) {
              console.log('删除失败');
            }
          });
        },
        error: function (object, error) {
          console.log("query object fail");
        }
      });
    }
  }
});
```



图 7-2 作业查看

7.1.5 交流区的实现

点击讨论区按钮进入聊天界面发布问题，然后提交。

7.2 老师模块的实现

7.2.1 老师模块的简述

老师可以查看所有学生的成绩，发布作业并且可以在探讨区和学生探讨。

7.2.2 发布作业模块的实现

老师点击发布作业按钮，进入作业发布界面，输入内容等信息之后提交，会在作业列表中显示，学生可以在列表中查看作业内容。核心代码：

```

//内部调用云端代码
    var currentUser = Bmob.User.current();
    var data = {
        "objectId": currentUser.id, "encryptedData": res.encryptedData, "iv":
res.iv
    };
    console.log(data);

    // console.log(data);
    Bmob.Cloud.run('getOpenGId', data).then(function (obj) {
        // var res = JSON.parse(obj)
        console.log(obj)
    }, function (err) {
        console.log(err)
    });

    data = { "objectId": currentUser.id, "encryptedData":
"Q3h+kMwbKZ52BsxdNT4GS5LTYeLLGIXnA/BZrg/9iMJBD5Qv3Fs5H66xe9ml7i
NIsOBetaeUG0InAxbZOhn1qEeAJ2aC3wYpjARR4pCYA1v87+bj9khaUDY6pvaK
X5/4TFHrofKAmA0gTT6bSaHyiw==", "iv": "YHoSkWomdfiyvAWhoYvKiQ==" };
    console.log(data);
    Bmob.Cloud.run('getOpenGId', data).then(function (obj) {
        // var res = JSON.parse(obj)
        console.log(obj)
    }, function (err) {
        console.log(err)
    });

```

7.2.3 交流区的实现

交流区和学生交流区共享，在此不做过多的介绍，直接展示核心代码：

```

onShareAppMessage: function () {
    var that = this;
    return {
        title: '课堂 聊天室',
        path: 'pages/yunyan/yunyan',
        success: function (res) {
            // 转发成功
            console.log('成功', res)
            var currentUser = Bmob.User.current();
            that.pushMessage(createSystemMessage('恭喜' +
currentUser.get('nickName') + '把聊天室成功分享给他人...'));
            wx.getShareInfo({
                shareTicket: res.shareTickets,
                success(res) {
                    fail: function (res) {
                        // 转发失败

```

8 系统测试

由于在需求分析、模块设计和编码实现的时候难免有一些缺陷和失误，即使系统可以运行起来，也会出现一些逻辑上的缺陷，因此对软件进行测试也是开发系统的一个必不可少的环节。如果不及时终止这些错误，最终轻则导致系统运行错误，重则导致程序崩溃。造成不必要的损失，浪费财力和物力。目前尚且没有彻底解决的办法避免错误的出现。所以软件测试是目前最有效的检测手段。

该教学管理平台的测试主要是针对功能和界面的测试：

界面测试是可以在不同的手机操作系统平台上进行测试，检测界面的兼容性。

功能测试是测试系统功能能否正常实现，逻辑是否正确，在逻辑测试方面主要是进行的是黑盒测试，若发现逻辑错误再进行白盒测试。

8.1 系统测试的意义

对系统进行检测可以检查出开发时候出现的问题，以及一些开发过程中没有注意的人为失误和逻辑上的错误。

测试的任务与目的：

目的：发现程序本身的错误和逻辑上的错误；

任务：通过多次数据的随机测试，尽量发现其中的错误。对一些有可能存在的问题，进行有针对性的测试。

另一个预测的相关术语叫纠错，其目的以及任务如下：

目的：定位并且纠正错误；

任务：解除软件的故障，从而保证程序的可靠运行。

如图 8-1 所示，从图中可以看出，测试需要多次的测试数据，这些测试数

据应该针对一些具有针对性的数据，这样才能检查出程序对各中数据是否正常处理。如果测试的数据与系统预期的结果相吻合，则说明基本无太大的问题，如果和预期出入太大，则说明有问题，需要检查代码和逻辑。

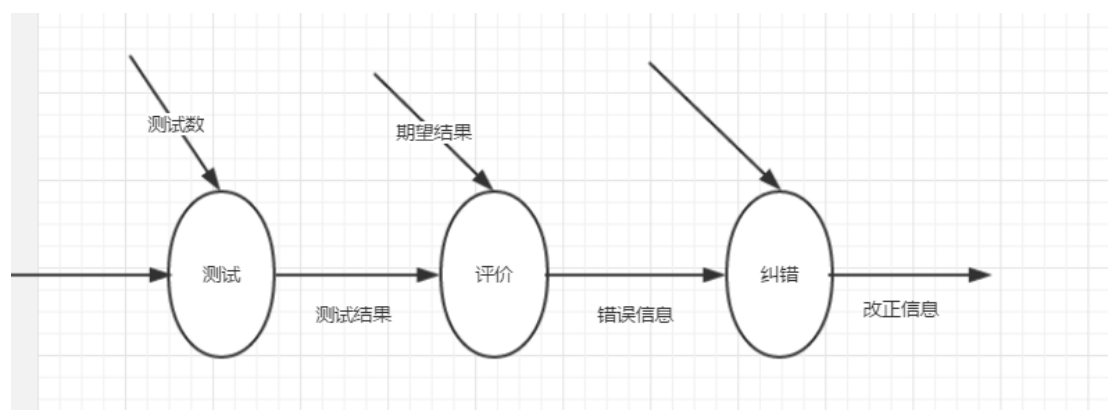


图 8-1 测试与纠错信息流程

8.2 具体测试任务

8.2.1 界面的测试

测试方法：在不同的手机平台上进行测试，当前主流的手机操作系统是 IOS 和安卓。分别在上述两个系统平台下的微信中测试，是否出现乱码和错误等一些界面问题。

测试结果：可以实现预期功能，截图如下：



图 8-2 界面实现

8.2.2 作业发布测试

测试方法：在发布作业的界面中随机输入不同的字符并且发布。检测是否可以在作业列表中显示。

测试结果：可以实现预期功能，截图如下：

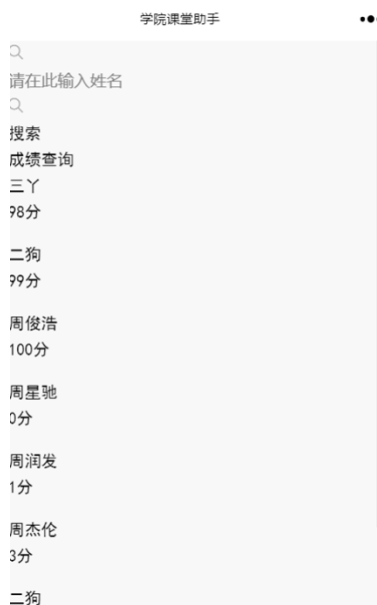


图 8-3 作业发布

8.2.3 成绩查询测试

测试方法：测试是否一个学生只能看到自己的成绩，但是老师可以看到所有人的成绩。

测试结果：可以实现预期功能，截图如下：



8-4 成绩部分

8.2.4 签到系统的测试

测试方法：检测学生事都可以正常签到，签到的结果（日期和时间，对于时间的显示必要性可以判断是否在规定的上课时间签到）是否可以在签到记录中显示。

测试结果：可以实现预期功能，截图如下：



8-5 签到系统

8.2.5 讨论区的测试

测试方法：交流的记录是否可以正常显示

测试结果：可以实现预期功能，截图如下：

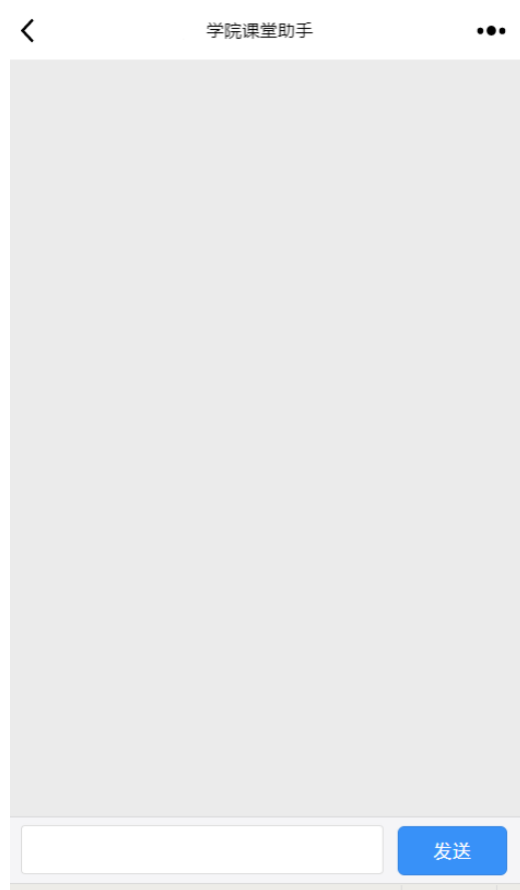


图 8-6 讨论区

8.3 测试分析

本师教学管理平台业务逻辑较为清晰，预期功能基本都可以实现，从设计到实现都基本都体现简洁的特点。教学管理平台有点描述如下：

- (1) 可移植性较强，在不同手机操作系统平台下都可以贴切手机尺寸。
- (2) 该教学管理平台内容全面，其操作方便，用户界面友好，在微信中可以获得较好的体验感。

经过对主要功能测试，可以满足预期功能。

结论

从 2017 年底提交开题报告，至今已经过去了 4 个多月。刚开始选题的时候，恰巧也是微信小程序最火的时候，当时设想能否把小程序不需要下载就可以应用这个功能用在师生交流方面，于是，就有设计基于微信小程序的教学管理平台的想法。

由于小程序刚出来，并且涉及到的技术都是全新的，要在短时间内做到掌握还是不容易的。在设计的过程中碰到问题都是前所未有的。

微信小程序总体上说是 JavaScript、CSS 和 HTML 三种综合的技术，把它们封装起来形成一种原生态的微信小程序语言。小程序原生态 API 接口是全新的接口，虽然官方提供了大量丰富的 API，但是它毕竟还是不成熟的。之前考虑能否上传视频，但是了解上传大小受限制等诸多不便因素在里面，最后改成论坛，以传统文字对话形式交流。

在开发这个教学管理平台时候，也慢慢感受到什么适合做微信小程序，什么不适合做。目前小程序在处理数据库方面及其不方便。它不能像 Java 一样，通过 JDBC 连接数据库，小程序只能发送 HTTP 请求。所以在设计教学管理平台时候，只能先通过 Web 端，然后再通过发送 request 请求访问数据库。目前微信小程序只能针对业务逻辑不是太复杂的应用。比如“饿了么”选择菜品、下单、付款。教学管理平台第一个版本功能简单完全试用。

在论文撰写的论文过程中遇到了很多难题。论文的撰写是一个系统工程，涉及到数据库，UML，软件工程等多门学科的交叉运用。由于以往课程忘记了不少，这些都变成论文撰写过程的拦路虎。特别是在用例图的制作上，以及 E-R 图的绘制。实体之间的联系分不清。

总体上问题得到了解决，也初步了解了微信小程序开发的过程。微信小程序的丰富的 API 函数尚且还有许多地方需要学习。如果要想学好微信小程序，JS、CSS 和 HTML 还是要好好学习，毕竟这三种开发语言才是小程序的灵魂所在。

参考文献

- [1] 黄曦, 沙拉依丁·苏里坦. 微信小程序开发快速入门[M]. 电子工业出版社, 2017.
- [2] 荣蓉, 穆心驰, 何金刚, 于连林. 实战微信小程序[M]. 电子工业出版社, 2017.
- [3] 雷磊. 微信小程序开发入门与实践[M]. 清华大学出版社, 2017.
- [4] 易伟. 微信小程序快速开发入门视频指导版[M]. 人民邮电出版社, 2017.
- [5] 张帆, 陈思含. 微信小程序开发零基础入门[M]. 电子工业出版社, 2017.
- [6] 王延平. 21天精通微信小程序开发[M]. 电子工业出版社, 2017.
- [7] 萨师煊, 王珊. 数据库系统概论(第三版)[M]. 北京: 高等教育出版社, 2004.
- [8] David Flanagan[美]. JavaScript权威指南[M]. 李强. 机械工业出版社, 2007.
- [9] 张伟. 基于JAVA的WEB在线考试系统[J]. 河南科技, 2014.
- [10] 李长强. 基于SSH2学生成绩管理系统的设计[J]. 电脑知识与技术, 2014.
- [11] 张焱, 党小超. 网络考试系统的设计与实现[N]. 甘肃广播电视大学学报, 2007.
- [12] 奥斯马尼[美]. JavaScript设计模式[M]. 徐涛. 人民邮电出版社, 2013.
- [13] 迈耶[美]. CSS权威指南(第3版)[M]. 尹志忠, 侯妍. 中国电力出版社, 2010.
- [14] 张鑫旭. CSS世界[M]. 人民邮电出版社, 2017.
- [15] 弗里曼[美]. HTML5权威指南. 谢廷晟. 人民邮电出版社, 2014.
- [16] 胡荷芬, 高斐. UML面向对象分析与设计教程[M]. 清华大学出版社, 2012.

致谢

大学四年即将结束。回想起四年前的自己，那时候刚踏入大学的校门，看到身边学长学姐们纷纷毕业，总感觉毕业离自己还有好久。如今自己正亲临这种感受，心头百感交集。这篇论文的结束，标志着自己大学四年本科学习和生活生涯走到了终点。在这里感谢曾经帮助过我的老师、亲人、同学和朋友们。

首先，感谢父母的养育之恩，以及支持我成年后自己做的每一个决定，没有你们的支持，我可能走不了这么远，飞不了那么高。

其次，真诚的感谢应文豪老师。应老师教授我的第一门课程是数据结构与算法。之后有幸参加应老师的项目，从中学到了很多知识。应老师待人和蔼可亲，每次请教应老师问题时候，老师都能耐心讲解。特别在我考研期间，给予了我莫大的关心和帮助。在此道一声：“老师，您辛苦了。”

然后，感谢我的两位班主任钱振江老师和王剑老师。钱振江老师是我大一到大三的班主任，平日里，钱老师和蔼可亲，特别上刚入学那会，在学业和生活上帮助我们给予我们许多帮助。王剑老师是我们大四时候的带班班主任。虽然只有短暂的一年接触。但是他特别的负责。特别在我们即将毕业的时候，好多事情他都会帮助我们处理。尽量减少我们的工作量。

之后，感谢我的同学们。很高兴在××××大学认识你们，在我困难的时候，你们能帮助我渡过难关，让我感受到集体的温暖，在一起四年很开心。

最后感谢母校××××大学。给我提供了舒适的生活环境，提供了许多教学资源，让我学到了很多专业知识。