**实验教学管理系统SDS设计分析**

**小组成员：2051840梁厚，2052528 杨淇深**

目录

[1. **引言** 2](#_Toc123323499)

[1.1 **编写目的** 2](#_Toc123323500)

[1.2 **背景及依据** 2](#_Toc123323501)

[2. **系统整体结构** 2](#_Toc123323502)

[2.1 **前后端框架以及技术栈** 2](#_Toc123323503)

[2.2 **代码分层设计** 4](#_Toc123323504)

[**前端层** 4](#_Toc123323505)

[**请求处理层** 4](#_Toc123323506)

[**业务逻辑层** 4](#_Toc123323507)

[**数据持久层** 4](#_Toc123323508)

[**内部数据源** 4](#_Toc123323509)

[3. **API设计** 5](#_Toc123323510)

[3.1 **用户管理模块** 5](#_Toc123323511)

[3.1.1 **用户登录验证API** 5](#_Toc123323512)

[3.1.2 **管理员批量导入教师虚拟账户API** 6](#_Toc123323513)

[3.1.3 **用户注册API** 7](#_Toc123323514)

[3.2 **教学管理模块** 8](#_Toc123323515)

[3.2.1 **设立课程API** 8](#_Toc123323516)

[3.2.2 **删除课程API** 8](#_Toc123323517)

[3.3 **教学资料管理模块** 9](#_Toc123323518)

[3.3.1 **上传教学资料API** 9](#_Toc123323519)

[3.4 **成绩管理模块** 10](#_Toc123323520)

[3.4.1 **上传成绩数据API** 10](#_Toc123323521)

[4. **数据库设计** 11](#_Toc123323522)

[4.1 **总体ER图** 11](#_Toc123323523)

[4.2 **用户管理模块** 12](#_Toc123323524)

[4.2.1 **实体-联系分析** 12](#_Toc123323525)

[4.2.2 **用户管理模块ER图** 13](#_Toc123323526)

[4.3 **教学管理模块** 13](#_Toc123323527)

[4.3.1 **实体-联系分析** 13](#_Toc123323528)

[4.3.2 **教学管理模块ER图** 14](#_Toc123323529)

[4.4 **教学资料管理模块** 14](#_Toc123323530)

[4.4.1 **实体-联系分析** 14](#_Toc123323531)

[4.4.2 **教学资料管理模块ER图** 14](#_Toc123323532)

[4.5 **成绩管理模块** 15](#_Toc123323533)

[4.5.1 **实体-联系分析** 15](#_Toc123323534)

[4.5.2 **成绩管理模块ER图** 15](#_Toc123323535)

[5. **类设计** 16](#_Toc123323536)

[5.1 **用户管理模块** 16](#_Toc123323537)

[5.1.1 **重要具体类详述** 17](#_Toc123323538)

[5.2 **教学管理模块** 36](#_Toc123323539)

[5.2.1 **重要具体类详述** 37](#_Toc123323540)

[5.3 **教学资料管理模块** 62](#_Toc123323541)

[5.3.1 **重要具体类详述** 64](#_Toc123323542)

[5.4 **成绩管理模块** 74](#_Toc123323543)

[5.4.1 **重要具体类详述** 76](#_Toc123323544)

1. **引言**

1.1 **编写目的**

本文档是用于细化实验教学管理系统的详细设计的，我们将通过**系统整体架构设计**、**API设计**、**数据库设计**、**类设计**4个角度规范实验教学管理平台开发中的编码方向、不同子模块之间的协作以及方法设计等。

由于系统开发过程中某些功能的开发、API、类设计逻辑上具有极大的相似性，所以本文档将选取具有代表性的代码和规范进行说明。

1.2 **背景及依据**

本项目名为**实验教学管理系统**，旨在创建一个实验教学管理系统，为参与实验课教学的教师、学生提供一个实验信息管理、实验报告上传和作业批改的平台。

本项目任务的提出者为软件工程课程设计需要；开发者为软件工程专业2名同学；项目预期用户是有实验教学管理需求的用户。

本文档的依据是**《实验教学管理系统SRS需求分析》**。即，本设计规约文档是基于需求分析的基础上进行编写的。

[实验教学管理系统SRS需求分析](https://ulqqdencc9.feishu.cn/docx/AV91d1AIoolubgxLRhiccXCxnyd)

2. **系统整体结构**

2.1 **前后端框架以及技术栈**

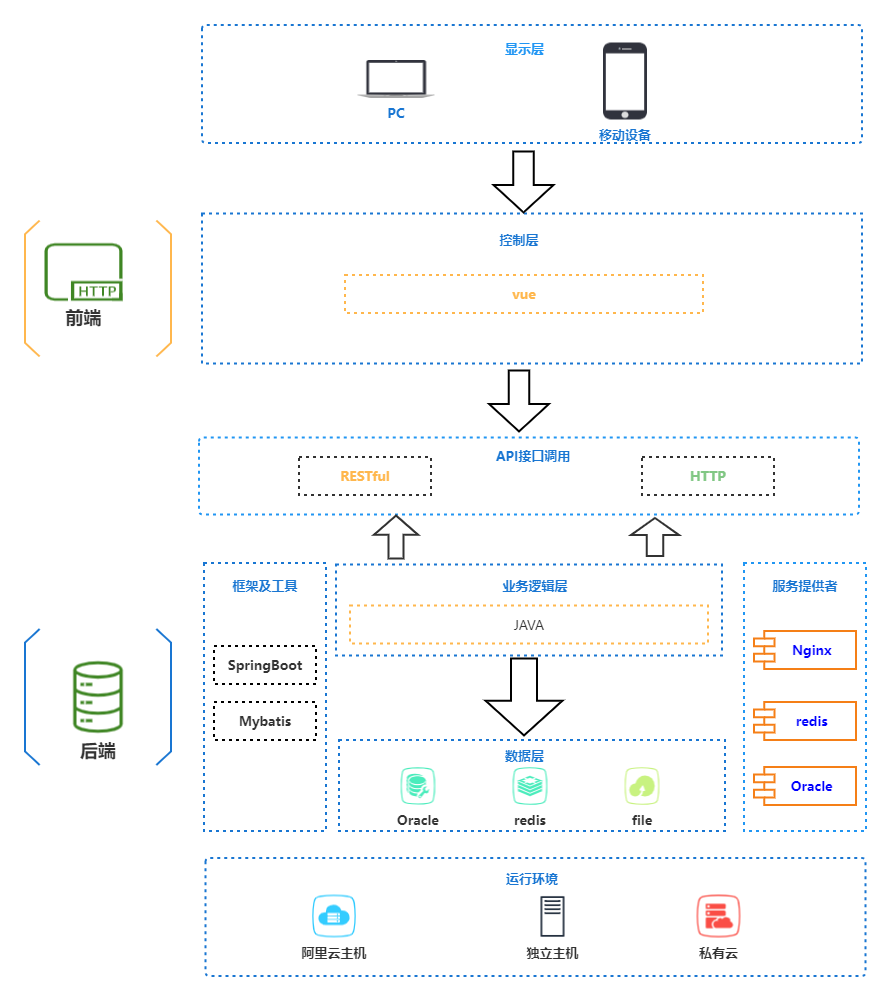
\*文字介绍\*

项目总体框架介绍：

项目采用前后端分离的方法搭建，即将一个单体应用拆分成两个独立的应用。前端项目使用vue框架，后端项目使用springboot框架。前后端通过Axios(Ajax)请求，RESTful风格API和HTTP协议进行通信。

* 后端在springboot框架中加入了Mybatis框架（数据持久层框架）与数据库进行交互。
* 使用oracle作为数据库存储数据。
* 数据库与前后端项目部署在阿里云ECS云服务器，利用阿里云OSS对象存储服务存放图片。

下面是总体框架概述图：



图表 1总体框架概述图

2.2 **代码分层设计**

项目代码采用分层的思想，在传统MVC架构的基础上将代码分成5层，分别是：前端层，请求处理层，业务逻辑层，数据持久层和内部数据源层。各层代码主要功能介绍如下：

**前端层**

前端层主要位于前端Vue项目，其的内容涵盖了从系统外发起的一切请求，还包括定时任务的触发配置，以及系统的监控和检测程序。

**请求处理层**

请求处理层的作用主要是封装RESTful API的HTTP接口（暴露给前端调用的各种类型的API）。

**业务逻辑层**

业务逻辑层的职责是与数据持久层交互，包括对多个数据源的操作进行聚合，并且提供组合复用的能力。此外，它也是业务通用能力的处理层，封装了请求处理层中对外暴露的接口的业务逻辑，而不是直接放在请求处理层的接口中，提高了安全性。业务逻辑层中还包括缓存方案、消息监听（MQ）、定时任务、参数校验、数据转换、异常处理等。

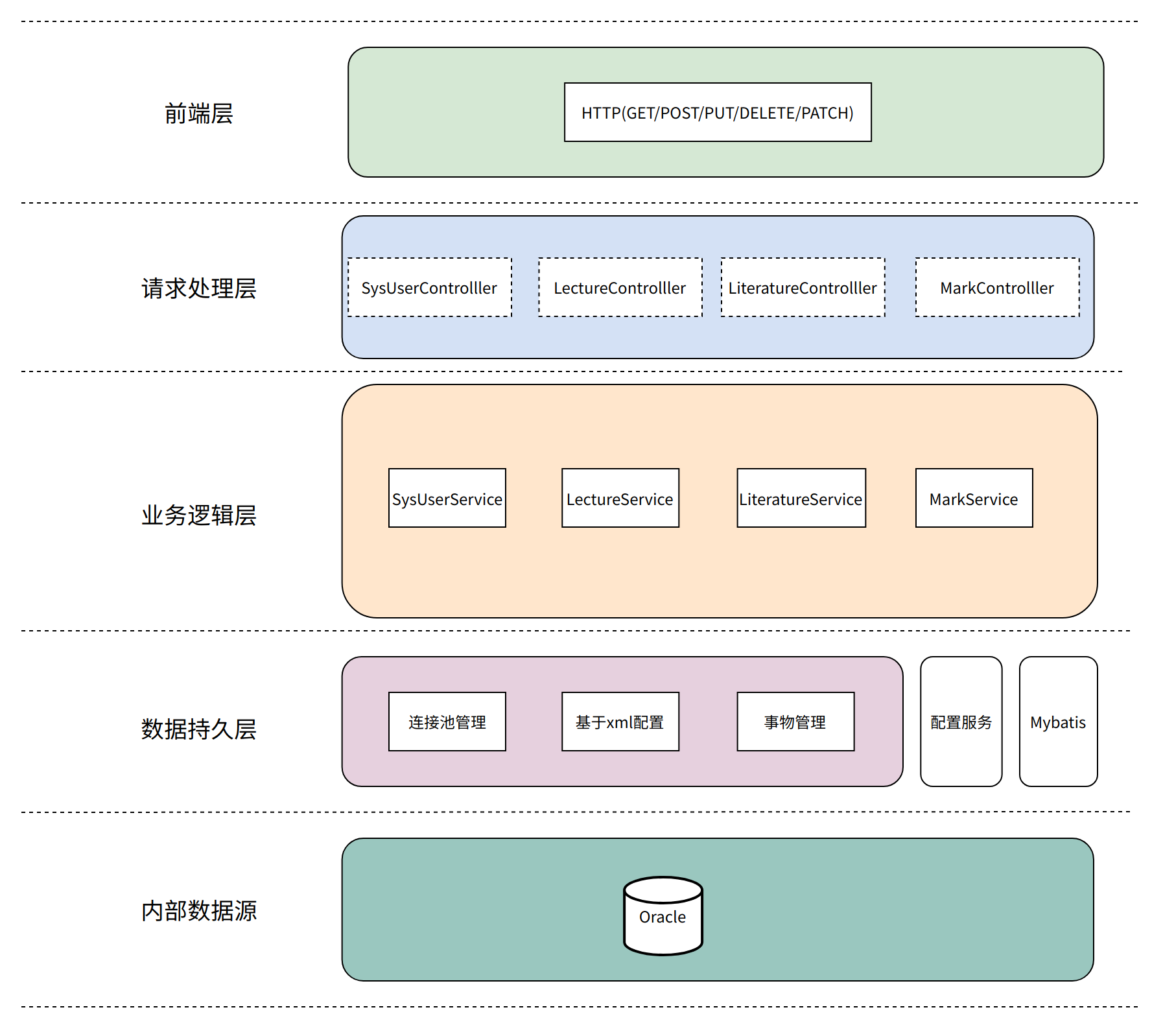
**数据持久层**

数据持久层承载了数据存储和访问的能力。包括通过DAO访问数据库的能力和通过DAP访问外部数据接口的能力。在我们的项目中，数据持久层主要提供了对单个数据库表的增删改查操作的接口，供业务逻辑层组合调用这些接口实现更复杂的逻辑。

**内部数据源**

内部数据源层中的每个类对应数据库中的一张表。

良好的代码分层设计可以使代码的耦合性大大降低，易用性大大增强，为系统的模块化拆分打下良好的基础。



图表 2代码分层设计概述图

3. **API设计**

我们项目采用RESTful风格的接口。RESTFUL是一种网络应用程序的设计风格和开发方式，**基于HTTP**，**可以使用 XML 格式定义或 JSON 格式定义**。最常用的数据格式是JSON。由于JSON能直接被JavaScript读取，所以，**使用JSON格式的REST风格的API具有简单、易读、易用的特点**。

REST 是面向资源的，每个资源都有一个唯一的资源定位符（URI）。每个URI代表一种资源（resource）。此外，设计的API将部署到服务器上，调用API时，下面的localhost需要改成服务器地址。

下面将举例系统4个子模块中具有代表性的API进行说明。

3.1 **用户管理模块**

3.1.1 **用户登录验证API**

* 接口描述：用于用户登录时进行验证
* 请求路径：[http://localhost:8080/user/v2/{email}/{password}](http://8.130.101.207:1827/api/Administrator/ReleaseNews)
* 请求方法：GET
* 示例：[http://localhost:8080/user/v2/1833150937@qq.com/123456](http://8.130.101.207:1827/api/Administrator/ReleaseNews)
* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
| email | string | 请求登录的用户的邮箱 | 无 |
| password | stirng | 用户现使用的密码 | 无 |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | String | True：验证成功；False：验证失败 |  |

响应示例：

|  |
| --- |
| C++ {  "code": 200,  "data": "True",  "msg": "success" } |

3.1.2 **管理员批量导入教师虚拟账户API**

* 接口描述：允许管理员向数据库中批量导入教师账户。管理员上传一个包含教师信息的excel表格文件，系统读取文件内容，写入数据库中。
* 请求路径：[http://localhost:8080/user/import\_tea/school\_id](http://localhost:8080/user/import_tea/1)
* 请求方法：POST
* 示例：http://localhost:8080/user/import\_tea/1
* 请求参数：
* 备注：默认插入的教师不是责任教师

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
| sys\_file | file | 教师信息excel文件 |  |
| school\_id | int | 导入账户的学校 |  |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | String | True：验证成功 | 注册失败会返回500 |

响应信息：

|  |
| --- |
| C++ {  "code": 200,  "data": "True",  "msg": "success" } |

3.1.3 **用户注册API**

* 接口描述：用户注册时，在数据库中添加新账户的信息。
* 请求路径：[http://localhost:8080/user/sys\_sign\_up/{email}/{password}](http://8.130.101.207:1827/api/Administrator/ReleaseNews)
* 请求方法：POST
* 示例：[http://localhost:8080/user/sys\_sign\_up/1306715855@qq.com/123456](http://localhost:8080/user/v3/1306715855@qq.com/123456)
* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明（前端） | 备注（后端） |
| email | string | 用户邮箱 |  |
| password | stirng | 用户输入的密码 |  |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | String | True：验证成功；False：验证失败 | 注册失败会返回500 |

3.2 **教学管理模块**

3.2.1 **设立课程API**

* 接口描述：提供（责任教师）设立新课程的API。
* 请求路径：[http://localhost:8080/lecture/set\_course/{cour\_name}/{res\_teacher\_id}](http://8.130.101.207:1827/api/Administrator/ReleaseNews)
* 请求方法：POST
* 示例：[http://localhost:8080/lecture/set\_course/嵌入式系统导论/97756](http://localhost:8080/lecture/v1/)
* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
| cour\_name | String | 课程名字 |  |
| res\_teacher\_id | int | 责任教师id |  |
| course\_outline | String | 课程大纲url地址 |  |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | string | True：设立成功；失败返回错误信息 |  |

3.2.2 **删除课程API**

* 接口描述：责任教师请求删除已设立的课程
* 请求路径：[http://localhost:8080/lecture/delete\_course/{course\_id}](http://localhost:8080/lecture/v5/96043)
* 请求方法：POST
* 示例：http://localhost:8080/lecture/delete\_course/6
* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明（前端） | 备注（后端） |
| course\_name | string | 课程名字 |  |
| res\_teacher\_id | int | 责任教师id |  |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | string |  |  |

响应信息：

|  |
| --- |
| Java {  "code": 200,  "data": "True",  "msg": "success" } |

3.3 **教学资料管理模块**

3.3.1 **上传教学资料API**

* 接口描述：提供上传教学资料的接口。数据库中保存资料文件对应的url地址。
* 请求路径：[http://localhost:8080/literature/upload/{name}/{exp\_id}/{literature\_url}/{school\_id}](http://localhost:8080/literature/upload/VLAN)
* 请求方法：POST
* 示例：[http://localhost:8080/literature/upload/VLAN介绍/6/test/1](http://localhost:8080/literature/upload/VLAN)
* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明（前端） | 备注（后端） |
| name | String | 上传的资料名称 |  |
| exp\_id | int | 所属实验Id |  |
| literature\_url | String | 文件的url地址 |  |
| school\_id | int | 所属学校Id |  |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | string | 成功返回True |  |

响应示例：

|  |
| --- |
| Java {  "code": 200,  "data": "True",  "msg": "success" } |

3.4 **成绩管理模块**

3.4.1 **上传成绩数据API**

* 接口描述：教师/助教批改报告后，记录批改的报告的成绩，并存入数据库。
* 请求路径：[http://localhost:8080/mark/set\_report\_mark/{exp\_id}/{student\_id}/{school\_id}/{mark}](http://localhost:8080/mark/set_report_mark/6/95001/1/8)
* 请求方法：POST
* 示例：http://localhost:8080/mark/set\_report\_mark/6/95001/1/8
* 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
| exp\_id | int | 报告对应的实验的Id |  |
| student\_id | int | 报告对应的学生的Id |  |
| school\_id | int | 报告对应的学校的Id |  |

* 响应参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数说明 | 说明 | 备注 |
|  | string | 上传成功返回“True” |  |

响应示例：

|  |
| --- |
| Java {  "code": 200,  "data": "True",  "msg": "success" } |

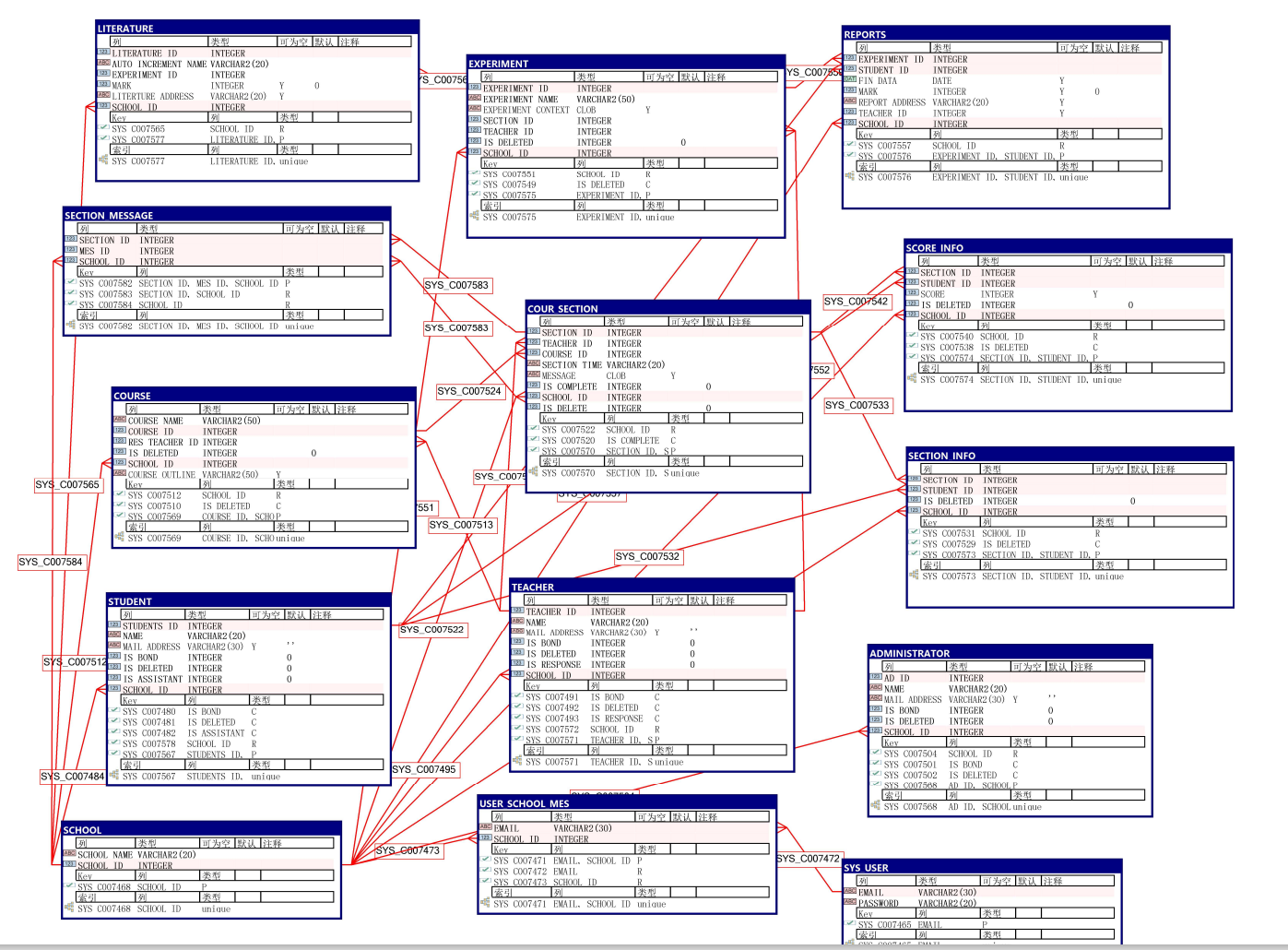
其他详细的API说明，见API文档：

[API文档](https://ulqqdencc9.feishu.cn/docx/HlLIdTuw6oD2zbxnoh3cPNn8nUe)

4. **数据库设计**

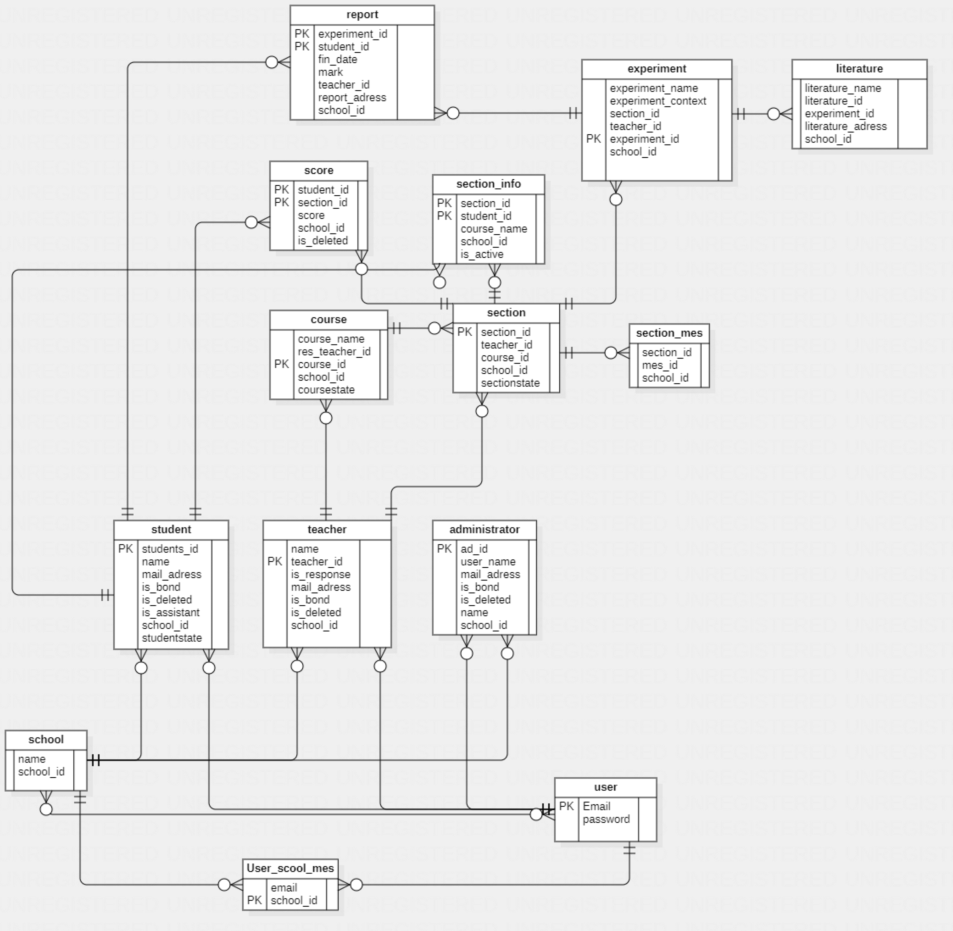
我们的数据库采用关系型数据库，工具采用oracle，把数据放入表格中，这些表格可以相互关联。

数据库中的表的关系图如下图所示：



图表 3数据库表关系图

4.1 **总体ER图**



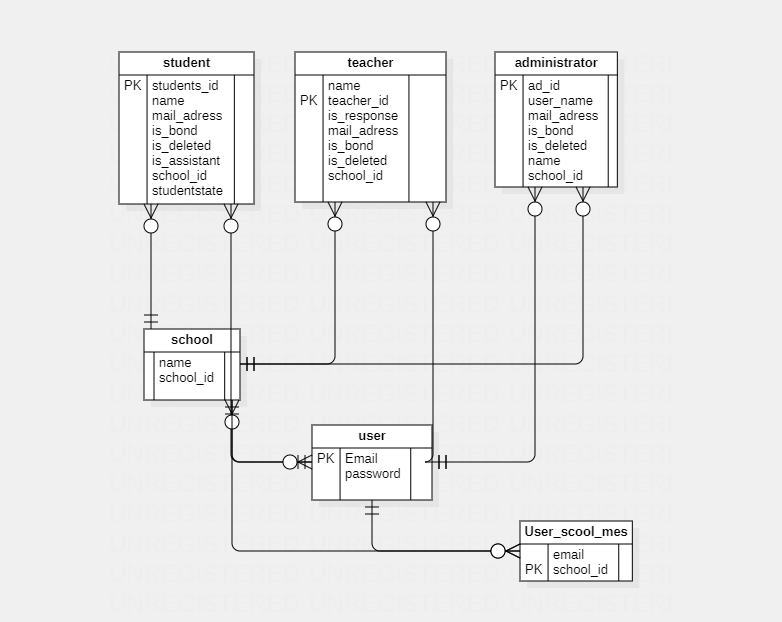
图表 4总体ER图

4.2 **用户管理模块**

4.2.1 **实体-联系分析**

1. 用户和学校（组织）的属于关系：用户可能属于多个不同的学校，一个学校有多个用户。例如：同济大学某学生小A属于同济大学，但是他和他的同伙在校外搞了一个神秘的实验组织。他们也需要用到教学实验管理平台。这里小A同学属于两个学校（组织），一个是同济大学，另一个是他和同伙的秘密地下组织。所以用户和学校是多对多关系。用户不一定全部参与学校（比如某用户小B只是进网站晃荡的无业游民，没有任何学校收留他）。
2. 用户和身份的绑定关系：用户在每一个学校里有且只有一个身份，而在其他学校他的身份可能发生变化。如在a学校的学生可能是b学校的老师。所以用户和身份（虚拟账户）的绑定关系属于一对多关系。用户可能有多种身份，也可能没有，而仅仅只是一个用户。
3. 学校和虚拟账户（老师、学生、管理员）的拥有关系：一个学校有多个学生、多个老师和多个管理员。在需求分析文档中提到同一个学校里的三种主要角色的账户属于同一个“用户群”。所以学校和虚拟账户的关系属于一对多关系。

4.2.2 **用户管理模块ER图**



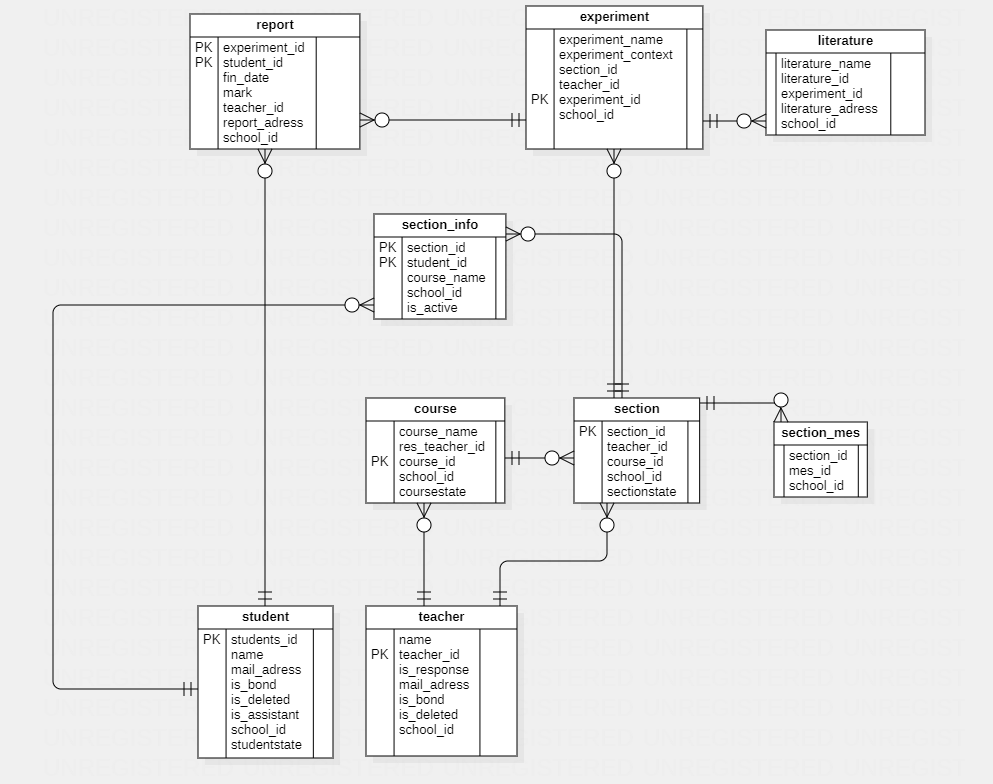
图表 5用户管理模块ER图

4.3 **教学管理模块**

4.3.1 **实体-联系分析**

1. course和section的对应关系：course是由责任教师发布的课程，而section是有具体任课老师和学生信息的实例。如：责任教师王某开设了计算机组成原理实验课，王某设置a老师开设并教授的A班，在周一-12节上课；同时王某又设置b老师开设并教授的B班，在周三-34节上课。a老师和b老师教授的课属于section，它们都属于计算机组成原理课，计算机组成原理课就对应数据库表中的course。所以course和section属于一对多关系。section全部参与课程，每一门课程至少有一个对应的section。
2. literature，report和experiment的从属关系：experiment是某一课程section中添加的实验，每个实验需要学生完成并提交报告；同时每个实验配套有相应的教学资料。即，每个实验experiment可以有多个literature或是report，而每一条literature和report对应一个experiment。实验和实验报告以及教学资料的关系是一对多关系。
3. 学生与课程的参与关系。学生参与相应课程其中一个section的授课。一门课由多个学生参与；一个学生参与多门课程。所以学生和课程的关系是多对多关系。关系表section\_info用于记录学生参与授课的情况。
4. 教师与课程的参与关系。责任教师设立课程，责任教师与课程course的关系是一对多关系；教师参与多门课的授课，而一门课程section由一名老师授课，所以教师与section的关系是一对多的关系。

4.3.2 **教学管理模块ER图**



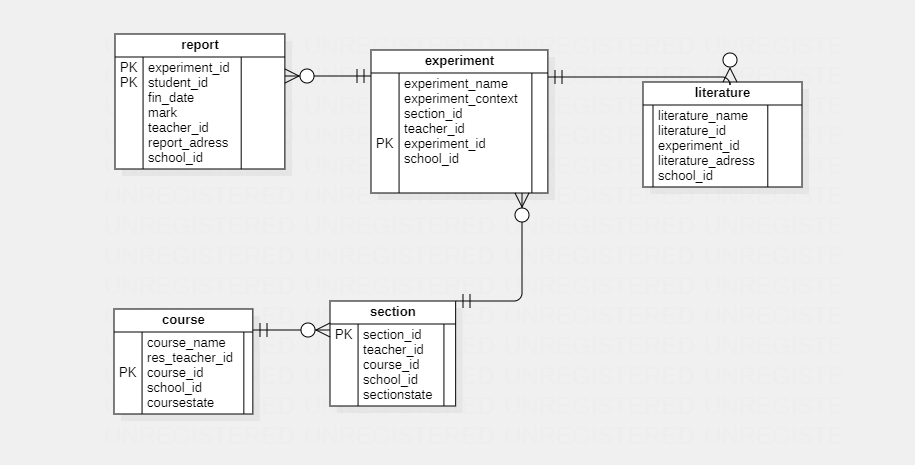
图表 6教学管理模块ER图

4.4 **教学资料管理模块**

4.4.1 **实体-联系分析**

1. 教学资料literature与实验的属于关系：教学资料以其描述的实验为依据进行分类管理。所以一个实验对应多个教学资料。实验和教学资料属于一对多关系。

4.4.2 **教学资料管理模块ER图**



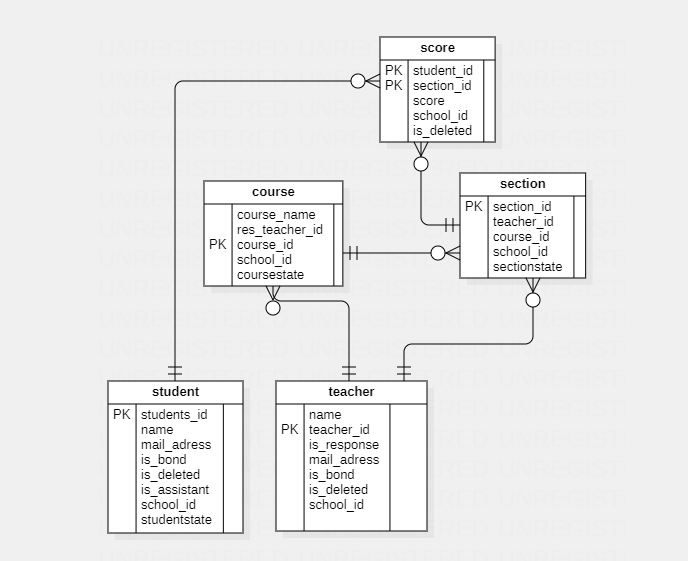
图表 7教学资料管理模块ER图

4.5 **成绩管理模块**

4.5.1 **实体-联系分析**

1. 学生和成绩的对应关系：每个学生参与的section都有与之对应的一个成绩。学生在section中的成绩用关系表score记录。所以学生和score是一对多关系。
2. score和mark的算数关系：score和mark是对于学生成绩的记录。mark是report的一个属性，标明了在某一个报告当中学生的得分，而score是对于该门课程所有实验mark的一个加权平均。

4.5.2 **成绩管理模块ER图**



图表 8成绩管理模块ER图

5. **类设计**

5.1 **用户管理模块**

**模块功能简述**

用户管理模块主要负责实现用户管理的相关功能。包括：

1. 用户注册。
2. （管理员）添加学生、教师信息。
3. 验证用户登录。
4. 修改账户信息。
5. 实现账户和虚拟账户绑定。

**类的协作关系概述**

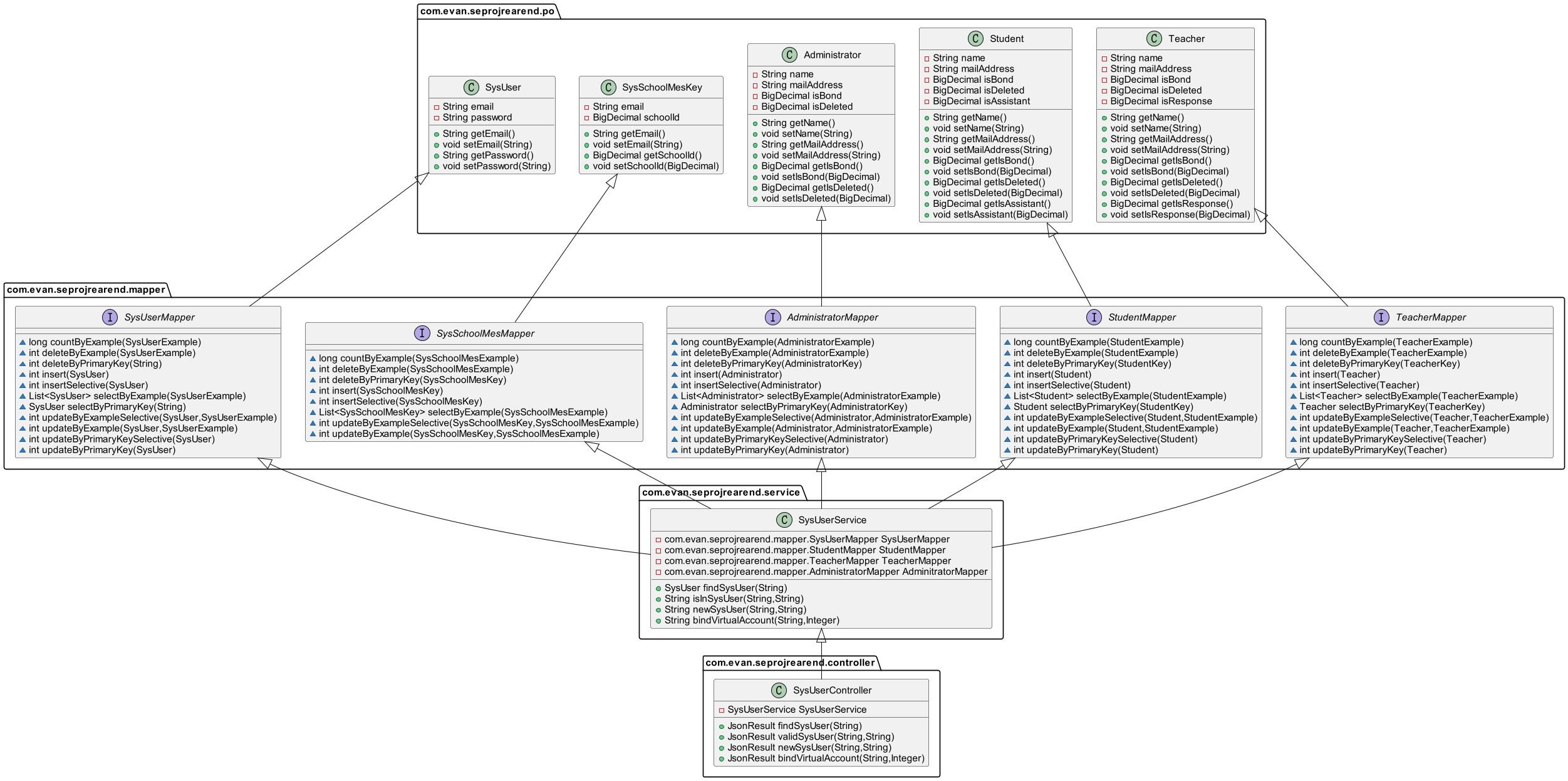
用户管理模块需要4个包中的类的协作。从底层到顶层依次为：po实体层，mapper数据持久层，service服务层，controller接口层。

po实体层中一共有5个实体类负责记录相关的用户信息。其中SysUser记录用户登录和使用系统的账户信息；SysSchoolMesKey类记录用户和学校系统之间的关系，其余Student，Teacher，Administrator对应每个学校系统中的角色，这3个记录特定的学校系统中与用户账户绑定的虚拟账户的相关信息。

每个实体类都有与之对应的mapper类实现基础的对数据的操作（增/删/改/查等）。

SysUserService类则通过调用mapper类中实现基础操作的函数，完成更加复杂的业务逻辑（即，对外调用的接口中复杂的业务逻辑在SysUserService类中实现）。SysUserController类则实现API供前端项目调用。

**用户管理模块类图**



图表 9用户管理模块类图

5.1.1 **重要具体类详述**

5.1.1.1 **SysUser类**

**功能**

对应数据库中SYS\_USER表。主要负责记录用户的邮箱信息和密码。向外提供获取和设置用户邮箱和密码的函数。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| email | String | 表示用户的邮箱 |
| password | String | 用户登录的密码 |

**方法描述**

**getEmail**

方法签名：public String getEmail()

功能说明：获取用户的邮箱并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该用户的email | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**getPassword**

方法签名：public String getPassword()

功能说明：获取用户的密码并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该用户的password | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setEmail**

方法签名：public void setEmail(String email)

功能说明：设置用户的邮箱。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 设置SysUser对象的邮箱 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 需要设置的该用户的email |

**setPassword**

方法签名：public void setPassword(String password)

功能说明：设置用户的密码。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 设置SysUser对象的密码 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| password | String | 需要设置的该用户的password |

5.1.1.2 **SysSchoolMesKey类**

**功能**

对应数据库中SYS\_SCHOOL\_MES表。主要负责记录用户与绑定系统的关系的信息。向外提供获取和设置参数的函数。例如，用户1可能在学校A中是学生，在学校B中是管理员。SysSchoolMesKey类则记录了这种用户到系统的映射关系。SysSchoolMesKey类的一个对象对应一条“用户-学校关系”。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| email | String | 用户与学校系统关系中用于标识用户的属性 |
| schoolId | BigDecimal | 学校的Id |

**方法描述**

**getEmail**

方法签名：public String getEmail()

功能说明：获取用户的邮箱并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该用户的email | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**getPassword()**

方法签名：public String getPassword()

功能说明：获取用户的密码并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该用户的password | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setEmail**

方法签名：public void setEmail(String email)

功能说明：设置用户的邮箱。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 设置新增关系中用户的邮箱 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 需要设置的该用户的email |

**setPassword**

方法签名：public void setPassword(String password)

功能说明：设置用户的密码。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 设置新增关系中学校的Id | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| password | String | 需要设置的该用户的password |

5.1.1.3 **Student类**

**功能**

对应数据库中STUDENT表。主要负责记录某学校系统中学生的信息。向外提供获取和设置参数的函数。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| name | String | 学生的姓名 |
| mailAddress | String | 学生邮箱的地址（作为绑定账户时验证的依据） |
| isBond | BigDecimal | 是否已经绑定具体的账户（0：未绑定；1：已绑定） |
| isDeleted | BigDecimal | 该学生信息是否已经删除（0：未删除；1：已删除） |
| isAssistant | BigDecimal | 该学生是否是助教（0：否；1：是） |

**方法描述**

**getMailAddress**

方法签名：public String getMailAddress()

功能说明：获取学生的邮箱并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该用户的email | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**getName**

方法签名：public String getName()

功能说明：获取学生的姓名并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该学生的name | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setMailAddress**

方法签名：public void setMailAddress(String mailAddress)

功能说明：设置学生的邮箱。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 设置Student对象的邮箱 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| mailAddress | String | 需要设置的该用户的email |

**setName**

方法签名：public void setName(String name)

功能说明：设置用户的密码。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 设置Student对象的姓名 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| name | String | 需要设置的该学生的姓名 |

说明：Student类的其他属性isBond, isDelete, isAssistant均有对应的get{属性名}和set{属性名}，其功能和使用方法和上述函数类似，在此就不一一举例。

此外，实体类中Teacher、Administrator类的功能和方法和Student类类似，此处略。

5.1.1.4 **SysUserMapper接口**

**功能**

实现对SysUser类属性值的基本的查找和修改操作。

**方法描述**

**countByExample**

方法签名：long countByExample(SysUserExample example)

功能说明：根据传入的SysUserExample对象，在数据库中查找与之匹配的记录的数目并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | Long | |
| 返回值 | 返回与传入的对象匹配的数据库中的记录的数目 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | SysUserExample | SysUserExample类的对象，调用时一般用new生成。 |

**deleteByExample**

方法签名：int deleteByExample(SysUserExample example)

功能说明：根据传入的SysUserExample对象，在数据库中查找与之匹配的记录并删去。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 删除成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | SysUserExample | SysUserExample类的对象，调用时一般用new生成。 |

**deleteByPrimaryKey**

方法签名：int deleteByPrimaryKey(String email)

功能说明：根据传入的用户邮箱email，在数据库中查找与之匹配的记录并删除。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 删除成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 希望删除的SysUser对象的主键email |

**insert**

方法签名：int insert(SysUser record)

功能说明：在数据库SYS\_USER表中插入传入的SysUser对象。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 插入成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | SysUser | SysUser类的对象 |

**insertSelective**

方法签名：int insertSelective(SysUser record)

功能说明：在数据库SYS\_USER表中插入传入的SysUser对象。与insert函数不同之处在于，insert函数中传入的SysUser对象必须具有SysUser类的全部属性，而insertSeletive可以部分置null或者使用默认值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 插入成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | SysUser | SysUser类的对象 |

**selectByExample**

方法签名：List<SysUser> selectByExample(SysUserExample example)

功能说明：根据传入的SysSchoolExample对象，在数据库中查找与之匹配的记录并返回符合条件的SysUser对象列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | List<SysUser> | |
| 返回值 | 返回与传入的对象匹配的数据库中的记录的对象列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | SysSchoolExample | SysSchoolExample类的对象 |

**selectByPrimaryKey**

方法签名：SysUser selectByPrimaryKey(String email)

功能说明：根据传入的email字符串，在数据库SYS\_USER表中查找主键与email一致的记录并返回单个SysUser对象。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | SysUser | |
| 返回值 | 返回主键与传入email一致的SysUser对象 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | SysUser类的对象 |

**updateByExampleSelective**

方法签名：int updateByExampleSelective(@Param("record") SysUser record**,** @Param("example") SysUserExample example)

功能说明：第一个参数record为修改的部分值组成的对象，第二个参数是一个对应的查询条件的类， 通过这个类可以实现order by 和一部分的where 条件。即，updateByExampleSelective函数将example中筛选出的所有记录的值更新为与record对象一致。updateByExampleSelective函数和updateByExample函数区别在于：updateByExample是传入一个对象,将整条数据都更新,如果对象中没有值的属性,就自动设置为null。updateByExampleSelective是将一行中某几个属性更新,而不改变其他的值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | SysUser | SysUser类的对象 |
| example | SysUserExample | SysUserExample类对象 |

**updateByExample**

方法签名：int updateByExample(@Param("record") SysUser record**,** @Param("example") SysUserExample example)

功能说明：第一个参数record为修改的部分值组成的对象，第二个参数是一个对应的查询条件的类， 通过这个类可以实现order by 和一部分的where 条件。即，updateByExampleSelective函数将example中筛选出的所有记录的值更新为与record对象一致。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | SysUser | SysUser类的对象 |
| example | SysUserExample | SysUserExample类对象 |

**updateByPrimaryKeySelective**

方法签名：int updateByPrimaryKeySelective(SysUser record)

功能说明：接收的参数为对应于数据库的实体类对象，利用字段的自动匹配进行更新表的操作，如果传入obj对象中的某个属性值为null，则不进行数据库对应字段的更新。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | SysUser | SysUser类的对象 |

**updateByPrimaryKey**

方法签名：int updateByPrimaryKey(SysUser record)

功能说明：接收的参数为对应于数据库的实体类对象，利用字段的自动匹配进行更新表的操作。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | SysUser | SysUser类的对象 |

说明：其他实体类的mapper类的函数与SysUserMapper类似，都是实现增删改查几个基本的对数据操作的功能，所以在此就不一一举例每个mapper的功能和内置的函数。

5.1.1.5 **SysUserService类**

**功能**

调用mapper接口实现间接的对数据库的操作，实现用户管理模块中所有接口的业务逻辑，为controller类提供服务。例如，实现用户登录验证的逻辑、用户注册新账户的逻辑以及用户和学校系统中的虚拟账户绑定等功能。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| SysUserMapper | SysUserMapper | 对SysUser类操作的接口类的对象 |
| StudentMapper | StudentMapper | 对Student类操作的接口类的对象 |
| TeacherMapper | TeacherMapper | 对Teacher类操作的接口类的对象 |
| AdminitratorMapper | AdminitratorMapper | 对Adminitrator类操作的接口类的对象 |
| SysSchoolMesMapper | SysSchoolMesMapper | 对SysSchoolMes类操作的接口类的对象 |

**方法描述**

**isInSysUser**

方法签名：public String isInSysUser(String email**,** String password)

功能说明：通过参数email和password与该用户的实际密码进行对比，查看密码是否正确。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 密码正确返回“True”，否则返回“False” | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 请求登录的用户的邮箱。 |
| password | String | 请求登录的用户输入的密码。 |

**newSysUser**

方法签名：public String newSysUser(String email**,**String password)

功能说明：用户注册时创建新账户。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 创建成功返回“True”，否则返回“False” | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 用户注册的邮箱。 |
| password | String | 用户设置的密码。 |

**importStudentByExcel**

方法签名：public String importStudentByExcel(List<SysMember> student\_list**,** Integer school\_id)

功能说明：（管理员）批量导入学生虚拟账户。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 导入成功返回“True” | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| student\_list | List<SysMember> | SysMember对象列表（用于存放读到的excel表格中的数据的类）。 |
| school\_id | Integer | 导入学校的Id。 |

**importTeacherByExcel**

方法签名：public String importTeacherByExcel(List<SysMember> teacher\_list**,** Integer school\_id)

功能说明：（管理员）批量导入教师虚拟账户。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 导入成功返回“True” | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| teacher\_list | List<SysMember> | SysMember对象列表（用于存放读到的excel表格中的数据的类）。 |
| school\_id | Integer | 导入学校的Id。 |

**bindVirtualAccount**

方法签名：public String bindVirtualAccount(String email**,** Integer school\_id)

功能说明：绑定用户账户和学校系统中的虚拟账户（添加“用户-学校”关系，修改虚拟账户中is\_bond的属性）。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 绑定成功返回“True”，否则返回“False” | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 请求绑定的用户的邮箱。 |
| school\_id | Integer | 导入学校的Id。 |

5.1.1.6 **SysUserController类**

**功能**

向外提供用户管理模块的相关接口。包括验证用户登录、用户注册新账户以及用户和学校系统中的虚拟账户绑定等。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| SysUserService | SysUserService | 用户管理模块中提供服务的类 |

**方法描述**

**validSysUser**

方法签名：public JsonResult validSysUser(@PathVariable String email**,**@PathVariable String password)throws Exception

功能说明：使用SysUserMapper的isInSysUser方法，验证用户登录的账号与密码是否一致。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 用户登录的账号与密码一致返回True，不一致返回False，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 用户登录账号 |
| password | String | 用户登录密码 |

**importStudentByExcel**

方法签名：public JsonResult importStudentByExcel(@RequestPart("sys\_file") MultipartFile sys\_file**,**@PathVariable Integer school\_id) throws IOException

功能说明：使用SysUserMapper的importStudentByExcel方法，使用excel表格批量导入学生账户。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 导入成功返回True，失败返回False，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sys\_file | MultipartFile | 需要导入的学生信息的excel文件 |
| school\_id | Integer | 导入的学校的Id |

**newSysUser**

方法签名：public JsonResult newSysUser(@PathVariable String email**,**@PathVariable String password)throws Exception

功能说明：使用SysUserMapper的newSysUser方法，创建新用户并加入到数据库中。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 新用户创建成功返回True，失败返回False，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 用户注册邮箱 |
| password | String | 用户注册密码 |

**importTeacherByExcel**

方法签名：public JsonResult importTeacherByExcel(@RequestPart("sys\_file") MultipartFile sys\_file**,**@PathVariable Integer school\_id) throws IOException

功能说明：使用SysUserMapper的importTeacherByExcel方法，批量导入教师账户。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 导入成功返回True，失败返回False，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sys\_file | MultipartFile | 需要导入的学生信息的excel文件 |
| school\_id | Integer | 导入的学校的Id |

**bindVirtualAccount**

方法签名：public JsonResult bindVirtualAccount(@PathVariable String email**,**@PathVariable Integer school\_id)throws Exception

功能说明：使用SysUserMapper的bindVirtualAccount方法，绑定用户与学校系统中的虚拟账户。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 用户登录的账号与密码一致返回True，不一致返回False，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| email | String | 用户登录账号 |
| password | String | 用户登录密码 |

5.2 **教学管理模块**

**模块功能简述**

教学管理模块主要负责实现教学管理相关的功能。包括：

1. 责任教师设立实验课程，修改已设立的实验课程的信息，删除课程。
2. 教师查看教授的课程的开课信息、课程大纲。
3. 教师根据课程的安排发布相应的实验。
4. 学生参与课程，查看课程中需要完成的实验的基本信息。
5. 学生完成实验后上传相应的实验报告。
6. 教师、助教查看学生上传的实验报告，并且批改报告，给出相应的成绩。
7. 学生查看已经批改的报告的成绩。

**类的协作关系概述**

教学管理模块需要4个包中的类的协作。从底层到顶层依次为：po实体层，mapper数据持久层，service服务层，controller接口层。

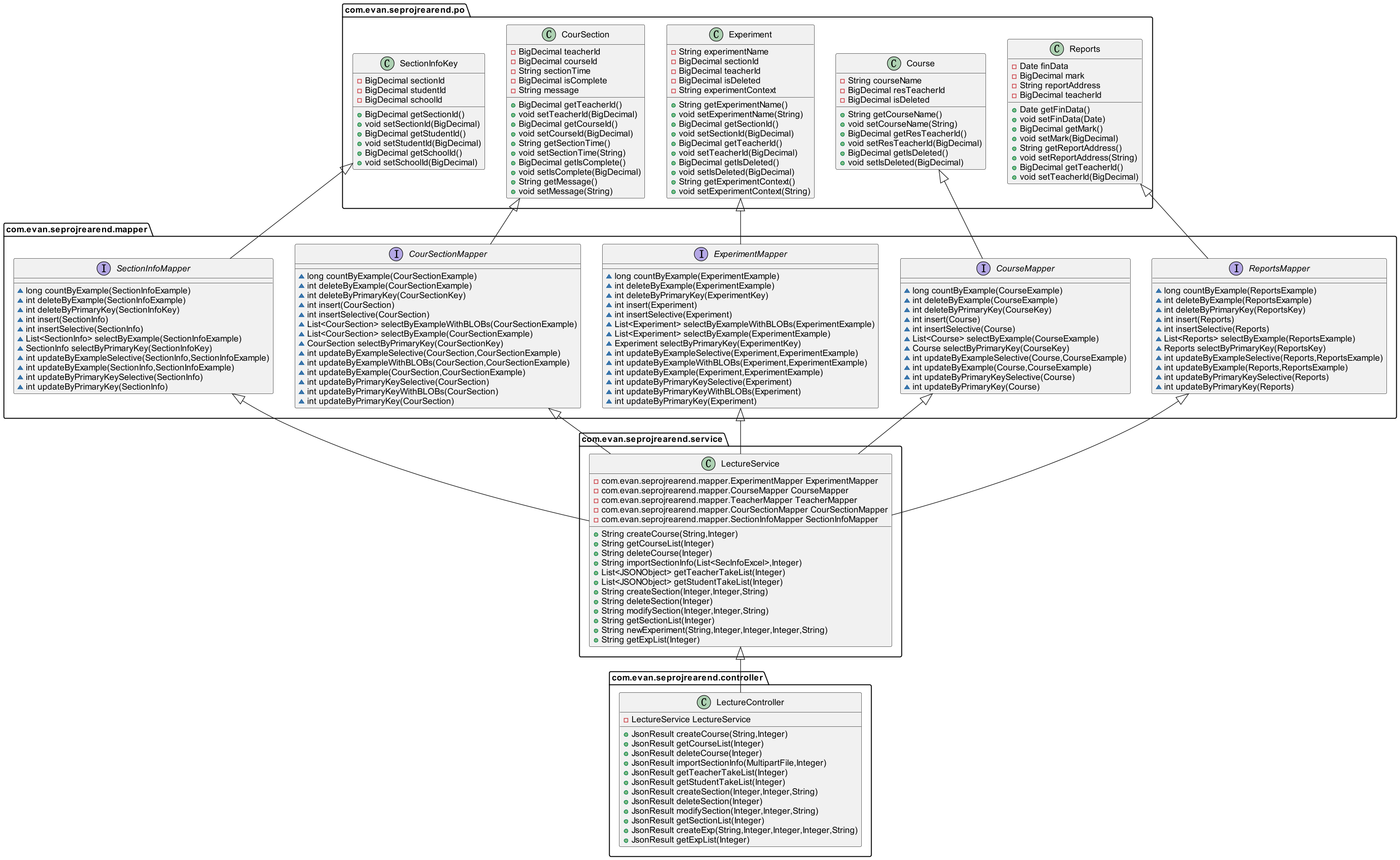
po实体层中一共有4个实体类负责记录相关的用户信息。由于课程管理模块主要管理课程-->具体课程-->实验

其中SysUser记录用户登录和使用系统的账户信息；SysSchoolMesKey类记录用户和学校系统之间的关系，其余Student，Teacher，Administrator对应每个学校系统中的角色，这3个记录特定的学校系统中与用户账户绑定的虚拟账户的相关信息。

每个实体类都有与之对应的mapper类实现基础的对数据的操作（增/删/改/查等）。

SysUserService类则通过调用mapper类中实现基础操作的函数，完成更加复杂的业务逻辑（即，对外调用的接口中复杂的业务逻辑在SysUserService类中实现）。SysUserController类则实现API供前端项目调用。

**教学管理模块类图**



图表 10教学管理模块类图

5.2.1 **重要具体类详述**

5.2.1.1 **Course类（√）**

**功能**

对应数据库中COURSE表。主要负责记录责任教师设立的课程的相关信息。向外提供获取课程相关信息参数的函数。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| courseName | String | 课程名称 |
| resTeacherId | BigDecimal | 责任教师Id |
| isDeleted | BigDecimal | 是否删除（0：未删除；1：已删除） |
| courseOutline | String | 课程大纲 |

**方法描述**

**getCourseName**

方法签名：public String getCourseName()

功能说明：返回课程名称。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该课程的名称。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setCourseName**

方法签名：public void setCourseName(String courseName)

功能说明：设置课程名称。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| courseName | String | 设置的课程的名称 |

**getResTeacherId**

方法签名：public BigDecimal getResTeacherId()

功能说明：返回责任教师Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该课程的责任教师Id。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setResTeacherId**

方法签名：public void setResTeacherId(BigDecimal resTeacherId)

功能说明：设置责任教师Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| resTeacherId | BigDecimal | 设置的责任教师的Id |

**getIsDeleted**

方法签名：public BigDecimal getIsDeleted()

功能说明：返回课程是否被删除的信息（0：未删除；1：已删除）

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回课程是否被删除的信息（0：未删除；1：已删除）。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setIsDeleted**

方法签名：public void setIsDeleted(BigDecimal isDeleted)

功能说明：删除课程。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| isDeleted | BigDecimal | 将isDeleted设为1时表示删除课程 |

**getCourseOutline**

方法签名：public String getCourseOutline()

功能说明：返回课程大纲

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回课程大纲url。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setCourseOutline**

方法签名：public void setCourseOutline(String courseOutline)

功能说明：设置课程大纲。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| courseOutline | String | 课程大纲url |

5.2.1.2 **SectionInfoKey类**

**功能**

对应数据库中SECTION\_INFO表。主要负责记录学生参与授课的情况。由于SECTION\_INFO表是关系表，所以SectionInfoKey类和实体表对应的类略有不同。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| sectionId | BigDecimal | 开课时段（section）Id |
| studentId | BigDecimal | 参与课程学生Id |
| schoolId | BigDecimal | 所属学校（机构）Id |

**方法描述**

**getSectionId**

方法签名：public BigDecimal getSectionId()

功能说明：返回课程时段的Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回课程时段的Id。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**getStudentId**

方法签名：public BigDecimal getStudentId()

功能说明：返回参与课程的学生Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回参与课程的学生Id。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**getSchoolId**

方法签名：public BigDecimal getSchoolId()

功能说明：返回所属学校/机构的Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回所属学校/机构的Id。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setStudentId**

方法签名：public void setStudentId(BigDecimal studentId)

功能说明：设置选课的学生的Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| studentId | BigDecimal | 设置的选课的学生的Id |

**setSchoolId**

方法签名：public void setSchoolId(BigDecimal schoolId)

功能说明：设置所属学校的Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| schoolId | BigDecimal | 设置的所属学校的Id |

**setSectionId**

方法签名：public void setSectionId(BigDecimal sectionId)

功能说明：设置课程时段（section）的Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sectionId | BigDecimal | 设置的课程时段（section）的Id |

5.2.1.3 **CourseMapper接口**

**功能**

实现对Course类属性值的基本的查找和修改操作。

**方法描述**

**countByExample**

方法签名：long countByExample(CourseExample example)

功能说明：根据传入的CourseExample对象，在数据库中查找与之匹配的记录的数目并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | Long | |
| 返回值 | 返回与传入的对象匹配的数据库中的记录的数目 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | CourseExample | CourseExample类的对象，调用时一般用new生成。 |

**deleteByExample**

方法签名：int deleteByExample(CourseExample example)

功能说明：根据传入的CourseExample对象，在数据库中查找与之匹配的记录并删去。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 删除成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | CourseExample | CourseExample类的对象，调用时一般用new生成。 |

**deleteByPrimaryKey**

方法签名：int deleteByPrimaryKey(CourseKey key)

功能说明：根据传入的CourseKey对象（由于COURSE表主键包含多个属性，CourseKey类用于标识CourseKey类的主键），在数据库中查找与之匹配的记录并删除。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 删除成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| key | CourseKey | 希望删除的Course对象的主键CourseKey的对象 |

**insert**

方法签名：int insert(Course record)

功能说明：在数据库COURSE表中插入传入的Course对象。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 插入成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Course | Course类的对象 |

**insertSelective**

方法签名：int insertSelective(Course record)

功能说明：在数据库COURSE表中插入传入的Course对象。与insert函数不同之处在于，insert函数中传入的Course对象必须具有Course类的全部属性，而insertSeletive可以部分置null或者使用默认值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 插入成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Course | Course类的对象 |

**selectByExample**

方法签名：List<SysUser> selectByExample(CourseExample example)

功能说明：根据传入的CourseExample对象，在数据库中查找与之匹配的记录并返回符合条件的Course对象列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | List<Course> | |
| 返回值 | 返回与传入的对象匹配的数据库中的记录的对象列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | CourseExample | CourseExample类的对象 |

**selectByPrimaryKey**

方法签名：SysUser selectByPrimaryKey(CourseKey key)

功能说明：根据传入的CourseKey对象，在数据库COURSE表中查找主键与CourseKey对象一致的记录并返回单个Course对象。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | Course | |
| 返回值 | 返回主键与传入CourseKey对象一致的Course对象 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| key | CourseKey | Course类的对象 |

**updateByExampleSelective**

方法签名：int updateByExampleSelective(@Param("record") SysUser record**,** @Param("example") CourseExample example)

功能说明：第一个参数record为修改的部分值组成的对象，第二个参数是一个对应的查询条件的类， 通过这个类可以实现order by 和一部分的where 条件。即，updateByExampleSelective函数将example中筛选出的所有记录的值更新为与record对象一致。updateByExampleSelective函数和updateByExample函数区别在于：updateByExample是传入一个对象,将整条数据都更新,如果对象中没有值的属性,就自动设置为null。updateByExampleSelective是将一行中某几个属性更新,而不改变其他的值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Course | Course类的对象 |
| example | CourseExample | CourseExample类对象 |

**updateByExample**

方法签名：int updateByExample(@Param("record") Course record**,** @Param("example") CourseExample example)

功能说明：第一个参数record为修改的部分值组成的对象，第二个参数是一个对应的查询条件的类， 通过这个类可以实现order by 和一部分的where 条件。即，updateByExampleSelective函数将example中筛选出的所有记录的值更新为与record对象一致。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Course | Course类的对象 |
| example | CourseExample | CourseExample类对象 |

**updateByPrimaryKeySelective**

方法签名：int updateByPrimaryKeySelective(Course record)

功能说明：接收的参数为对应于数据库的实体类对象，利用字段的自动匹配进行更新表的操作，如果传入obj对象中的某个属性值为null，则不进行数据库对应字段的更新。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Course | Course类的对象 |

**updateByPrimaryKey**

方法签名：int updateByPrimaryKey(Course record)

功能说明：接收的参数为对应于数据库的实体类对象，利用字段的自动匹配进行更新表的操作。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Course | Course类的对象 |

说明：其他实体类的mapper类的函数与SysUserMapper类似，都是实现增删改查几个基本的对数据操作的功能，所以在此就不一一举例每个mapper的功能和内置的函数。

5.2.1.4 **LectureService类**

**功能**

调用mapper接口实现间接的对数据库的操作，实现教学管理模块中所有接口的业务逻辑，为controller类提供服务。例如，创建新课程、返回学生参与的课程列表和教师创建新实验等功能。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| ExperimentMapper | ExperimentMapper | 对Experiment类操作的接口类的对象 |
| CourseMapper | CourseMapper | 对Course类操作的接口类的对象 |
| TeacherMapper | TeacherMapper | 对Teacher类操作的接口类的对象 |
| CourSectionMapper | CourSectionMapper | 对CourSection类操作的接口类的对象 |
| SectionInfoMapper | SectionInfoMapper | 对SectionInfo类操作的接口类的对象 |

**方法描述**

**createCourse**

方法签名：public String createCourse(String cour\_name**,** Integer res\_teacher\_id)

功能说明：创建新课程，并插入数据库中。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 设立成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| cour\_name | String | 设立的课程的名称 |
| res\_teacher\_id | Integer | 设立课程责任教师Id |

**getCourseList**

方法签名：public String getCourseList(Integer res\_id) throws JsonProcessingException

功能说明：返回（责任教师）设立的课程列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回责任教师设立的课程列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| res\_id | Integer | 责任教师Id |

**deleteCourse**

方法签名：public String deleteCourse(Integer course\_id)

功能说明：删除（责任教师）创建的课程（修改课程的isDelete属性）。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 删除成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| course\_id | Integer | 需要删除的课程的Id |

**importSectionInfo**

方法签名：public String importSectionInfo(List<SecInfoExcel> sec\_info\_list**,** Integer school\_id)

功能说明：批量导入学生的选课信息。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 导入成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sec\_info\_list | List<SecInfoExcel> | SecInfoExcel对象的列表（SecInfoExcel对象是读取选课信息excel表格信息并存储的类） |
| school\_id | Integer | 所属学校Id |

**getTeacherTakeList**

方法签名：public List<JSONObject> getTeacherTakeList(Integer teacher\_id)

功能说明：返回教师任课的课程相关信息列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | List<JSONObject> | |
| 返回值 | 返回教师任课的课程相关信息列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| teacher\_id | Integer | 教师Id |

**getStudentTakeList**

方法签名：public List<JSONObject> getStudentTakeList(Integer student\_id)

功能说明：返回学生参与的课程相关信息列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | List<JSONObject> | |
| 返回值 | 返回学生参与的课程相关信息列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| student\_id | Integer | 学生Id |

**createSection**

方法签名：public String createSection(Integer course\_id**,** Integer teacher\_id**,** String Section\_time)

功能说明：创建新的课程开课时段，并插入数据库中。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 设立成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| Section\_time | String | 授课时间 |
| teacher\_id | Integer | 任课教师Id |
| course\_id | Integer | 开课对应的课程Id |

**deleteSection**

方法签名：public String deleteSection(Integer sec\_id)

功能说明：删除开课时段（section）。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 删除成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sec\_id | Integer | 删除的section的Id |

**modifySection**

方法签名：public String modifySection(Integer sec\_id**,**Integer teacher\_id**,**String sec\_time)

功能说明：修改开课时段（section）。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 修改成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sec\_id | Integer | 修改的section的Id |
| teacher\_id | Integer | 修改后的任课教师Id |
| sec\_time | String | 修改后的上课时段 |

**getSectionList**

方法签名：public String getSectionList(Integer course\_id) throws JsonProcessingException

功能说明：返回某课程所有开课时段的列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回某课程所有开课时段的列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| course\_id | Integer | 课程的Id |

**newExperiment**

方法签名：public String newExperiment(String exp\_name**,**Integer sec\_id**,**Integer teacher\_id**,**Integer school\_id**,**String exp\_content)

功能说明：发布新实验。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 新实验发布成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| exp\_name | String | 实验名称 |
| sec\_id | Integer | 实验所属开课时段的Id |
| teacher\_id | Integer | 开设实验的教师的Id |
| school\_id | Integer | 开设新实验的学校的Id |
| exp\_content | String | 实验内容 |

**getExpList**

方法签名：public String getExpList(Integer sec\_id) throws JsonProcessingException

功能说明：获取某section下的实验列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 删除成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sec\_id | Integer | 删除的section的Id |

5.2.1.5 **LectureController类**

**功能**

向外提供教学管理模块的相关接口。包括创建新课程、返回学生参与的课程列表和教师创建新实验等。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| LectureService | LectureService | 教学管理模块中提供服务的类 |

**方法描述**

**createCourse**

方法签名：public JsonResult createCourse(@PathVariable String cour\_name**,**@PathVariable Integer res\_teacher\_id)

功能说明：（责任教师）设立课程。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该课程的名称，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| cour\_name | String | 该函数不需要参数 |
| res\_teacher\_id | Integer | 责任教师Id |

**getCourseList**

方法签名：public JsonResult getCourseList(@PathVariable Integer res\_teacher\_id)

功能说明：返回（责任教师）设立的课程列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 返回责任教师设立的课程列表的json字符串，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| res\_teacher\_id | Integer | 责任教师Id |

**deleteCourse**

方法签名：public JsonResult deleteCourse(@PathVariable Integer course\_id)

功能说明：删除（责任教师）创建的课程（修改课程的isDelete属性）。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 删除成功返回True，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| course\_id | Integer | 需要删除的课程的Id |

**importSectionInfo**

方法签名：public JsonResult importSectionInfo(@RequestPart("info\_file") MultipartFile info\_file**,** @PathVariable Integer school\_id)

功能说明：批量导入学生的选课信息。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 成功返回True，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| info\_file | MultipartFile | 传入的excel文件 |
| school\_id | Integer | 责任教师Id |

**getTeacherTakeList**

方法签名：public JsonResult getTeacherTakeList(@PathVariable Integer teacher\_id)

功能说明：返回教师任课的课程相关信息列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 返回教师任课的课程相关信息列表的json字符串，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| teacher\_id | Integer | 教师Id |

**getStudentTakeList**

方法签名：public JsonResult getStudentTakeList(@PathVariable Integer student\_id)

功能说明：返回学生参与的课程相关信息列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 返回学生参与的课程相关信息列表，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| student\_id | Integer | 学生Id |

**createSection**

方法签名：public JsonResult createSection(@PathVariable Integer course\_id**,**@PathVariable Integer teacher\_id**,**@PathVariable String section\_time)

功能说明：创建新的课程开课时段。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 成功返回True，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| Section\_time | String | 授课时间 |
| teacher\_id | Integer | 任课教师Id |
| course\_id | Integer | 开课对应的课程Id |

**modifySection**

方法签名：public JsonResult modifySection(@PathVariable Integer section\_id**,**@PathVariable Integer teacher\_id**,**@PathVariable String section\_time)

功能说明：修改开课时段（section）的相关数据。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 成功返回True，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| section\_id | Integer | 修改的section的Id |
| teacher\_id | Integer | 修改后的任课教师Id |
| section\_time | String | 修改后的上课时段 |

**deleteSection**

方法签名：public JsonResult deleteSection(@PathVariable Integer section\_id)

功能说明：删除开课时段（section）。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 成功返回True，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| sec\_id | Integer | 删除的section的Id |

**getSectionList**

方法签名：public JsonResult getSectionList(@PathVariable Integer course\_id)

功能说明：返回某课程所有开课时段的列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 返回某课程所有开课时段的列表，异常返回**10001,**"未知错误" | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| course\_id | Integer | 课程的Id |

**createExp**

方法签名：public JsonResult createExp(@PathVariable String experiment\_name**,**@PathVariable Integer section\_id**,**@PathVariable Integer teacher\_id**,**@PathVariable Integer school\_id**,**@PathVariable String exp\_content)throws Exception

功能说明：（任课教师）发布新实验。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回该课程的名称。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| experiment\_name | String | 实验名称 |
| section\_id | Integer | 实验所属开课时段的Id |
| teacher\_id | Integer | 开设实验的教师的Id |
| school\_id | Integer | 开设新实验的学校的Id |
| exp\_content | String | 实验内容 |

5.3 **教学资料管理模块**

**模块功能简述**

教学资料管理模块主要应当完成以下功能需求：

1. 教师可以上传教学资料。
2. 教师、学生可以下载教学资料。

**类的协作关系概述**

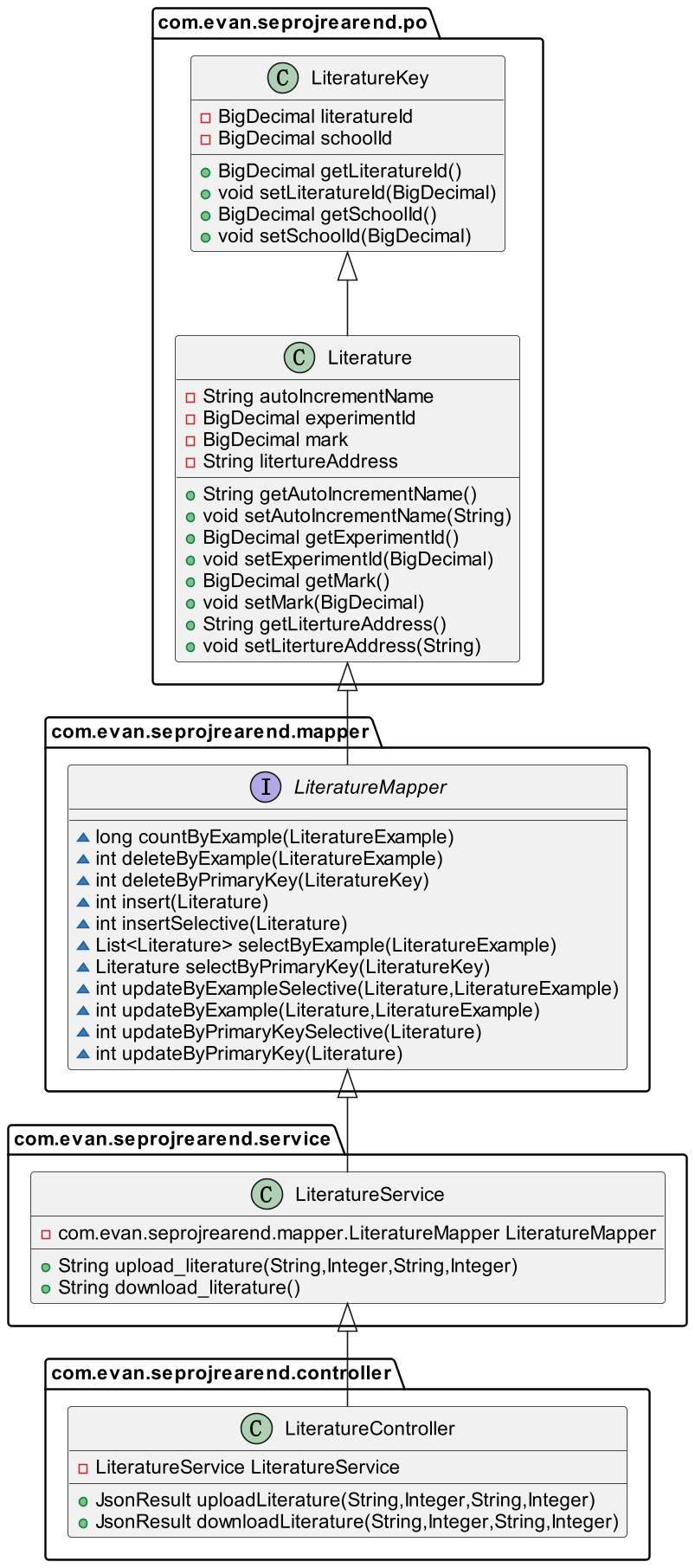
教学管理模块需要4个包中的类的协作。从底层到顶层依次为：po实体层，mapper数据持久层，service服务层，controller接口层。

po实体层中有1个实体类Literature负责记录教学资料的相关信息。

和之前的模块类似Literature有与之对应的mapper接口类实现基础的对数据的操作（增/删/改/查等）—— LiteratureMapper接口。

LiteratureService类则通过调用mapper类中实现基础操作的函数，完成复杂的业务逻辑。LiteratureController类则实现API供前端项目调用。

**教学资料管理模块类图**



图表 11教学资料管理模块类图

5.3.1 **重要具体类详述**

5.3.1.1 **Literature类**

**功能**

对应数据库中LITERATURE表。主要负责记录上传的教学资料等相关信息。向外提供获取和设置属性值的函数。是LiteratureKey的派生类。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| autoIncrementName | String | 资料名称 |
| experimentId | BigDecimal | 资料所属实验Id |
| litertureAddress | String | 资料的url地址 |

**方法描述**

**getAutoIncrementName**

方法签名：public String getAutoIncrementName()

功能说明：获取资料名称。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 返回资料名称。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setAutoIncrementName**

方法签名：public void setAutoIncrementName(String autoIncrementName)

功能说明：设置资料名称。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| autoIncrementName | String | 设置的资料名称 |

Literature类中其他属性均有一个get{属性名}() 函数和一个set{属性名}() 函数，它们的逻辑和功能类似，在此就不一一举例。

5.3.1.2 **LiteratureKey类**

**功能**

对应数据库中LITERATURE表的主键。向外提供获取和设置属性值的函数。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| literatureId | BigDecimal | 资料Id |
| schoolId | BigDecimal | 资料所属学校Id |

**方法描述**

**getLiteratureId**

方法签名：public BigDecimal getLiteratureId()

功能说明：返回资料Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回资料Id | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**getSchoolId**

方法签名：public BigDecimal getSchoolId()

功能说明：返回学校Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回学校Id | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setSchoolId**

方法签名：public void setSchoolId(BigDecimal schoolId)

功能说明：设置学校Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| schoolId | BigDecimal | 设置的学校Id |

**setLiteratureId**

方法签名：public void setLiteratureId(BigDecimal literatureId)

功能说明：设置资料Id。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | void | |
| 返回值 | 无 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| literatureId | BigDecimal | 设置的资料Id |

5.3.1.3 **LiteratureMapper接口**

**功能**

实现对Literature类和LiterautreKey类属性值的基本的查找和修改操作。

**方法描述**

**countByExample**

方法签名：long countByExample(LiteratureExample example)

功能说明：根据传入的CourseExample对象，在数据库中查找与之匹配的记录的数目并返回。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | Long | |
| 返回值 | 返回与传入的对象匹配的数据库中的记录的数目 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | LiteratureExample | LiteratureExample类的对象，调用时一般用new生成。 |

**deleteByExample**

方法签名：int deleteByExample(LiteratureExample example)

功能说明：根据传入的LiteratureExample对象，在数据库中查找与之匹配的记录并删去。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 删除成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | LiteratureExample | LiteratureExample类的对象，调用时一般用new生成。 |

**deleteByPrimaryKey**

方法签名：int deleteByPrimaryKey(LiteratureKey key)

功能说明：根据传入的CourseKey对象（由于COURSE表主键包含多个属性，CourseKey类用于标识CourseKey类的主键），在数据库中查找与之匹配的记录并删除。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 删除成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| key | LiteratureKey | 希望删除的Literature对象的主键LiteratureKey的对象 |

**insert**

方法签名：int insert(Literature record)

功能说明：在数据库LITERATURE表中插入传入的Literature对象。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 插入成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Literature | Literature类的对象 |

**insertSelective**

方法签名：int insertSelective(Literature record)

功能说明：在数据库LITERATURE表中插入传入的Literature对象。与insert函数不同之处在于，insert函数中传入的Literature对象必须具有Literature类的全部属性，而insertSeletive可以部分置null或者使用默认值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 插入成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Literature | Literature类的对象 |

**selectByExample**

方法签名：List<Literature> selectByExample(LiteratureExample example)

功能说明：根据传入的CourseExample对象，在数据库中查找与之匹配的记录并返回符合条件的Course对象列表。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | List<Literature> | |
| 返回值 | 返回与传入的对象匹配的数据库中的记录的对象列表 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| example | LiteratureExample | LiteratureExample类的对象 |

**selectByPrimaryKey**

方法签名：Literature selectByPrimaryKey(LiteratureKey key)

功能说明：根据传入的LiteratureKey对象，在数据库LITERATURE表中查找主键与LiteratureKey对象一致的记录并返回单个Literature对象。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | Course | |
| 返回值 | 返回主键与传入LiteratureKey对象一致的Literature对象 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| key | LiteratureKey | LiteratureKey类的对象 |

**updateByExampleSelective**

方法签名：int updateByExampleSelective(@Param("record") Literature record**,** @Param("example") LiteratureExample example)

功能说明：第一个参数record为修改的部分值组成的对象，第二个参数是一个对应的查询条件的类， 通过这个类可以实现order by 和一部分的where 条件。即，updateByExampleSelective函数将example中筛选出的所有记录的值更新为与record对象一致。updateByExampleSelective函数和updateByExample函数区别在于：updateByExample是传入一个对象,将整条数据都更新,如果对象中没有值的属性,就自动设置为null。updateByExampleSelective是将一行中某几个属性更新,而不改变其他的值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Literature | Literature类的对象 |
| example | LiteratureExample | LiteratureExample类对象 |

**updateByExample**

方法签名：int updateByExample(@Param("record") Literature record**,** @Param("example") LiteratureExample example)

功能说明：第一个参数record为修改的部分值组成的对象，第二个参数是一个对应的查询条件的类， 通过这个类可以实现order by 和一部分的where 条件。即，updateByExampleSelective函数将example中筛选出的所有记录的值更新为与record对象一致。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Literature | Literature类的对象 |
| example | LiteratureExample | LiteratureExample类对象 |

**updateByPrimaryKeySelective**

方法签名：int updateByPrimaryKeySelective(Literature record)

功能说明：接收的参数为对应于数据库的实体类对象，利用字段的自动匹配进行更新表的操作，如果传入obj对象中的某个属性值为null，则不进行数据库对应字段的更新。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Literature | Literature类的对象 |

**updateByPrimaryKey**

方法签名：int updateByPrimaryKey(Literature record)

功能说明：接收的参数为对应于数据库的实体类对象，利用字段的自动匹配进行更新表的操作。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | int | |
| 返回值 | 更新成功返回1；否则返回exception | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| record | Literature | Literature类的对象 |

5.3.1.4 **LiteratureService类**

**功能**

调用mapper接口实现间接的对数据库的操作，实现教学资料管理模块中所有接口的业务逻辑，为controller类提供服务。例如，实现上传、下载教学资料的功能。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| LiteratureMapper | LiteratureMapper | 对Literature类操作的接口类的对象 |

**方法描述**

**upload\_literature**

方法签名：public String upload\_literature(String name**,**Integer exp\_id**,**String literature\_url**,**Integer school\_id)

功能说明：允许用户上传教学资料。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 上传成功，返回“True”。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| name | String | 上传的资料的名字 |
| exp\_id | Integer | 资料所属实验的Id |
| literature\_url | String | 资料的url地址 |
| school\_id | Integer | 学校Id |

5.3.1.5 **LiteratureController类**

**功能**

向外提供教学资料管理模块的相关接口。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| LiteratureService | LiteratureService | 教学资料管理模块中提供服务的类 |

**方法描述**

**uploadLiterature**

方法签名：public JsonResult uploadLiterature(@PathVariable String name**,**@PathVariable Integer exp\_id**,**@PathVariable String literature\_url**,**@PathVariable Integer school\_id)

功能说明：通过参数pwd与该用户的实际密码进行对比，查看密码是否正确。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 上传成功，返回“True”。 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| name | String | 上传的资料的名字 |
| exp\_id | Integer | 资料所属实验的Id |
| literature\_url | String | 资料的url地址 |
| school\_id | Integer | 学校Id |

5.4 **成绩管理模块**

**模块功能简述**

教学管理模块主要负责实现教学管理相关的功能。包括：

1. 助教批改成绩，将成绩上传。
2. 教师、学生查看已经批改的报告的成绩。
3. 每个学生的成绩可以作为文档导出。

**类的协作关系概述**

教学管理模块需要4个包中的类的协作。从底层到顶层依次为：po实体层，mapper数据持久层，service服务层，controller接口层。

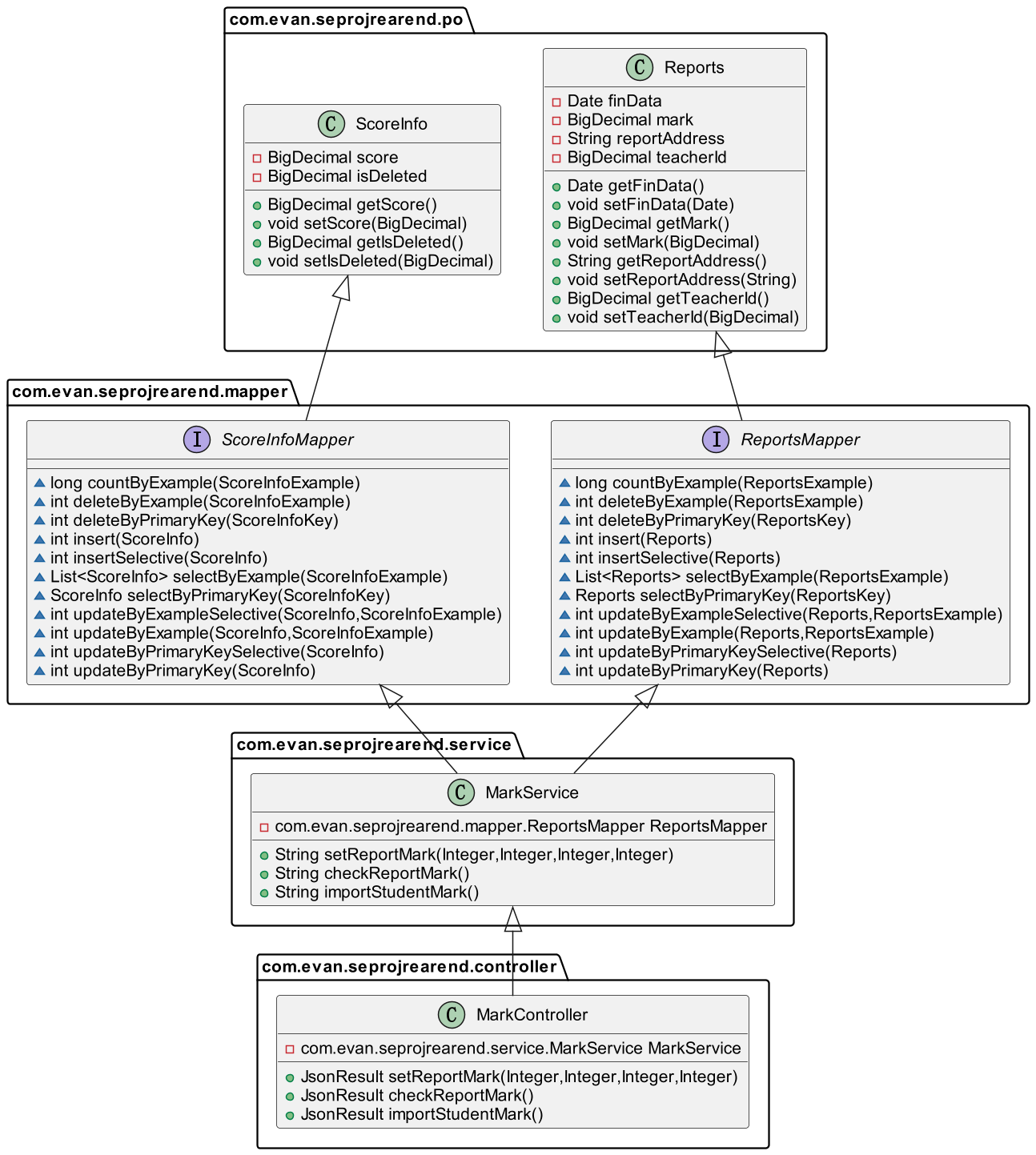
po实体层中一共有2个实体类负责记录相关的用户信息。

其中ScoreInfo记录学生在某个课程section中实验的最终得分；Reports类中mark属性记录学生在报告中的得分。

每个实体类都有与之对应的mapper类实现基础的对数据的操作（增/删/改/查等）。

MarkService类则通过调用mapper类中实现基础操作的函数，完成更加复杂的业务逻辑（即，对外调用的接口中复杂的业务逻辑在MarkService类中实现）。MarkController类则实现API供前端项目调用。

**成绩管理模块类图**



图表 12成绩管理模块类图

5.4.1 **重要具体类详述**

5.4.1.1 **ScoreInfo类**

**功能**

对应数据库中SCOREINFO表。主要负责记录学生的课程总成绩。向外提供获取和设置属性值的函数。是ScoreInfoKey的派生类。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| score | BigDecimal | 学生对应section的总成绩 |
| isDeleted | BigDecimal | 标识该记录是否被删除（0：未删除；1：已删除） |

**方法描述**

**getScore**

方法签名：public BigDecimal getScore()

功能说明：返回学生该section总成绩score的值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 返回学生该section总成绩score的值 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
|  |  | 该函数不需要参数 |

**setScore**

方法签名：public void setScore(BigDecimal score)

功能说明：返回学生该section总成绩score的值。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | BigDecimal | |
| 返回值 | 设置学生该section总成绩score的值 | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| score | BigDecimal | 学生成绩 |

5.4.1.2 **ScoreInfoMapper接口**

**功能**

实现对ScoreInfoMapper类属性值的基本的查找和修改操作。

**方法描述**

这里的Mapper接口类对表的增删改查函数和之前的类一致，在此不做过多描述。

5.4.1.3 **MarkService类**

**功能**

调用mapper接口实现间接的对数据库的操作，实现成绩管理模块中所有接口的业务逻辑，为controller类提供服务。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| ReportsMapper | ReportsMapper | 对Reports类操作的接口类的对象 |
| ScoreInfoMapper | ScoreInfoMapper | 对ScoreInfo类操作的接口类的对象 |

**方法描述**

**setReportMark**

方法签名：public String setReportMark(Integer exp\_id**,**Integer student\_id**,**Integer school\_id**,**Integer mark)

功能说明：上传报告成绩。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | String | |
| 返回值 | 成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| exp\_id | Integer | 报告对应的实验的Id |
| student\_id | Integer | 报告对应学生的Id |
| school\_id | Integer | 报告对应学校的Id |
| mark | Integer | 报告对应分数 |

5.4.1.4 **MarkController类**

**功能**

向外提供成绩管理模块的相关接口。

**属性描述**

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 说明 |
| ReportsMapper | ReportsMapper | 对Reports类操作的接口类的对象 |
| ScoreInfoMapper | ScoreInfoMapper | 对ScoreInfo类操作的接口类的对象 |

**方法描述**

**setReportMark**

方法签名：public JsonResult setReportMark(@PathVariable Integer exp\_id**,**@PathVariable Integer student\_id**,**@PathVariable Integer school\_id**,**@PathVariable Integer mark)

功能说明：上传报告成绩。

详细说明：

|  |  |  |
| --- | --- | --- |
| 返回类型 | JsonResult | |
| 返回值 | 成功返回True | |
| 方法限定符 | public | |
| 参数列表（参数名 / 类型 / 说明） | | |
| exp\_id | Integer | 报告对应的实验的Id |
| student\_id | Integer | 报告对应学生的Id |
| school\_id | Integer | 报告对应学校的Id |
| mark | Integer | 报告对应分数 |

图表索引

[图表 1总体框架概述图 4](#_Toc123324374)

[图表 2代码分层设计概述图 6](#_Toc123324375)

[图表 3数据库表关系图 12](#_Toc123324376)

[图表 4总体ER图 13](#_Toc123324377)

[图表 5用户管理模块ER图 14](#_Toc123324378)

[图表 6教学管理模块ER图 15](#_Toc123324379)

[图表 7教学资料管理模块ER图 16](#_Toc123324380)

[图表 8成绩管理模块ER图 17](#_Toc123324381)

[图表 9用户管理模块类图 19](#_Toc123324382)

[图表 10教学管理模块类图 39](#_Toc123324383)

[图表 11教学资料管理模块类图 65](#_Toc123324384)

[图表 12成绩管理模块类图 78](#_Toc123324385)