



# 北京航空航天大学

BEIHANG UNIVERSITY

## 数据库系统原理课程设计

### J 务管理系统

#### 系统实现报告

组	员	一	吕云帆	18373763
组	员	二	马瑞	17005044
组	员	三	马逸行	18182648

## 一、实现环境

系统使用 django 框架实现，完全使用 django 实现前后端的交互，前后端协同开发。系统实现基于 MTV 设计模式，即模型、模板、视图。

模型 (Model)：数据存取层。数据库使用 MySQL 数据库，而数据库相关操作则使用 django 提供的 API，使用自带 ORM 实现数据库表项与 Python 类之间的映射。

模板 (Template)：表现层。前端使用 HTML+CSS+JS 的模式，主要表现内容由 Template 中的 HTML 文件决定。

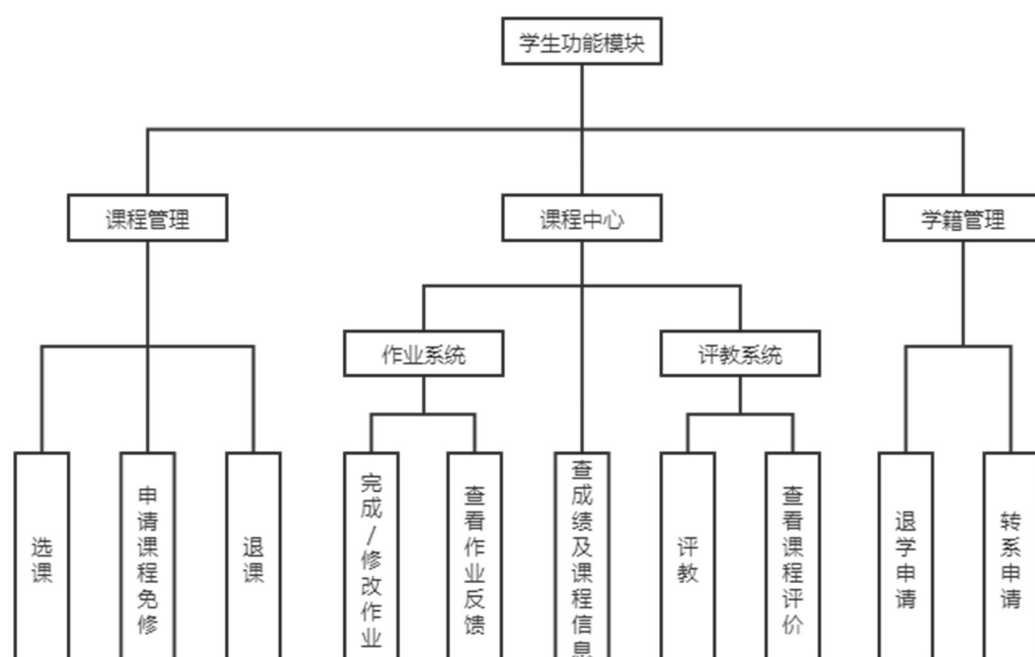
视图 (View)：业务逻辑层。后端的核心，接受前端传来的数据后进行处理，在数据层进行相关操作后做出反馈，反映到表现层上，是模型与模板的桥梁。

## 二、系统功能结构图

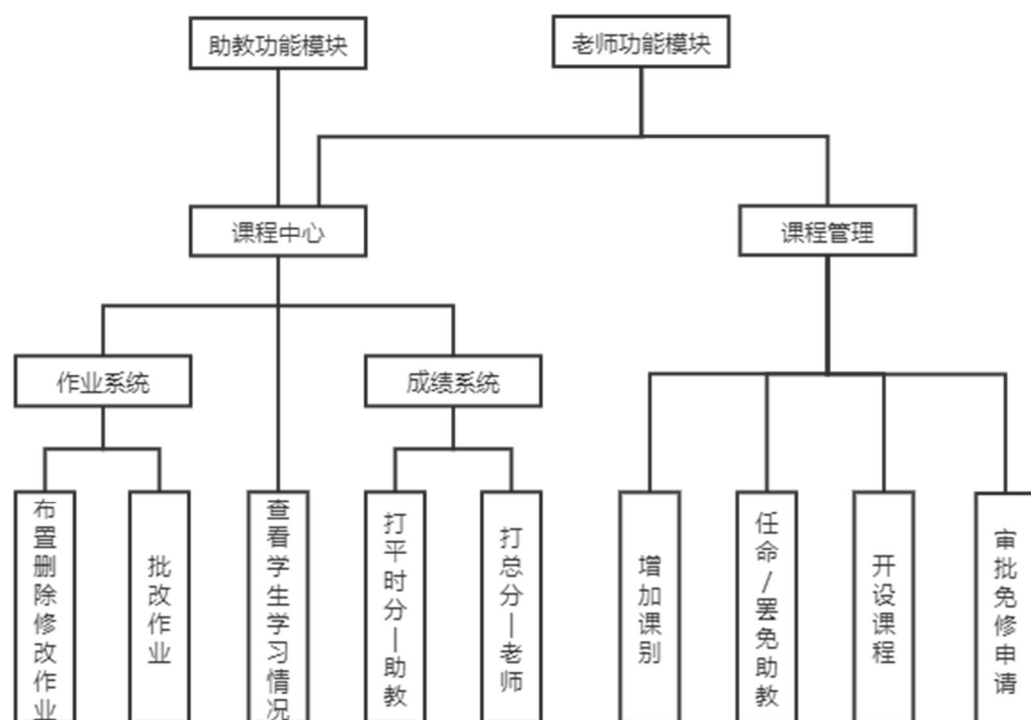
总览：



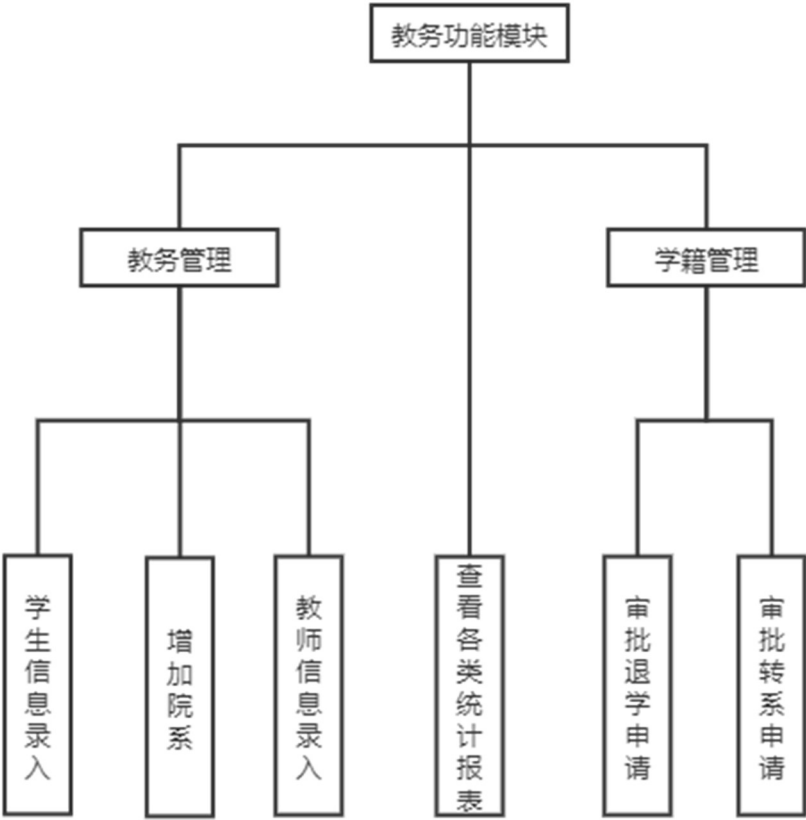
学生功能模块：



老师、助教功能模块：



教务功能模块：



### 三、 基本表、完整性约束和索引的定义

#### 1. 基本表和主键

基本表的定义如下：

(1) 学生表 Student

序号	名称	数据类型	主键	外键	功能
1	Sno	int	是		学号
2	Sname	char[20]			学生姓名
3	Dno	int		Department	所属院系
4	pw	char[100]			加密后密码

(2) 老师表 Teacher

序号	名称	数据类型	主键	外键	功能
1	Tno	int	是		工号
2	Tname	char[20]			老师姓名
3	pw	char[100]			加密后密码

(3) 院系表 Department

序号	名称	数据类型	主键	外键	功能
1	Dno	int	是	否	系号
2	Dname	char[20]	否	否	系名

(4) 课程表 Course

序号	名称	数据类型	主键	外键	功能
1	Cno	int	是		课程号
2	Cname	char[20]			课程名
3	credit	int			学分
4	v	int			容量

(5) 开课表 TC

序号	名称	数据类型	主键	外键	功能
1	Tno	int	是	是	开课老师
2	Dno	int		是	开课院系
3	Cno	int		是	课程号
4	s	int			已选人数

(6) 选课表 SC

序号	名称	数据类型	主键	外键	功能
1	Sno	int	是	是	选课学生
2	TC	int		是	所选课程
3	free	tinyint			是否免修
4	grade	int			总分
5	end	tinyint			是否结课
6	daily	int			平时分
7	dailyend	tinyint			是否已打平时分

(7) 作业表 HW

序号	名称	数据类型	主键	外键	功能
1	id	int	是		作业编号
2	TC	int		是	作业课程
3	name	char[30]			作业标题
4	question	char[50]			作业内容
5	times	int			提交上限
6	c	int			已提交人数

(8) 作业提交表 HWD

序号	名称	数据类型	主键	外键	功能
1	HW	int	是	是	提交作业号
2	Sno	int		是	提交学生
3	content	char[50]			作答内容
4	read	tinyint			是否已阅
5	point	int			得分
6	back	char[50]			反馈
7	had	int			提交次数

(9) 助教表 Tutor

序号	名称	数据类型	主键	外键	功能
1	TC	int	是	是	助教课程
2	Sno	int		是	助教学号

(10)免修申请表 FreeApply

序号	名称	数据类型	主键	外键	功能
1	id	int	是		申请号
2	SC	int		是	申请学生及课程
3	read	tinyint			是否已审批
4	accept	tinyint			是否同意
5	reason	char[30]			申请原因

(11)评教信息表 Remark

序号	名称	数据类型	主键	外键	功能
1	SC	int	是	是	评教学生及课程
2	p0	int			评分 0
3	p1	int			评分 1
4	p2	int			评分 2
5	p3	int			评分 3
6	p4	int			评分 4
7	p5	int			评分 5
8	r	char[50]			评论
9	rec	tinyint			是否推荐

### (12)退学申请表 Drop

序号	名称	数据类型	主键	外键	功能
1	id	int	是		申请号
2	Sno	int		是	申请学生
3	read	tinyint			是否已审批
4	accept	tinyint			是否同意
5	reason	char[30]			申请原因

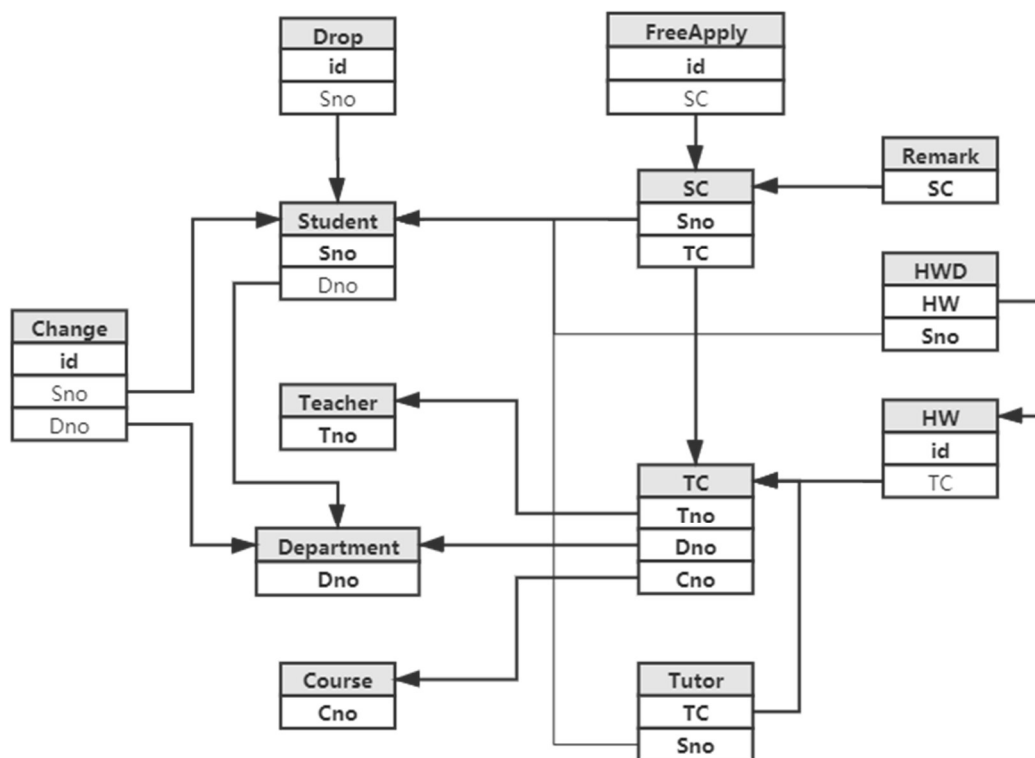
### (13)转系申请表 Change

序号	名称	数据类型	主键	外键	功能
1	id	int	是		申请号
2	Sno	int		是	申请学生
3	Dno	int		是	目标院系
4	read	tinyint			是否已审批
5	accept	tinyint			是否同意
6	reason	char[30]			申请原因

## 2. 完整性约束

实体完整性约束与参照完整性约束分别体现之前表里的“主键”与“外键”项

里，现画出实体关系图如下，更为直观：



自定义完整性如下：

- ◆ 不特殊说明的情况下，所有属性均不能为空；
- ◆ Department 表的 Dname 唯一；
- ◆ Course 表的 Cname 唯一，credit 是一个 0-10 的整数；
- ◆ SC 表的 end、daily 取值均为 0-100 的整数；
- ◆ HW 的 name 唯一，times 为非负整数；
- ◆ HWD 的 point 为 0-10 的浮点数；
- ◆ Remark 的 p0-5 均为 0-5 的整数。

自定义完整性的实现除了在 MySQL 数据库的表上增加约束外，还利用 django 的 ORM 约束，以及后端接收数据后利用条件语句判断。

### 3. 索引

索引使用 MySQL 由主键生成的索引，除此之外，数据库中没有其他索引。

## 四、 系统的安全性、人员权限

系统的安全性由前后端共同实现。在登录时共用一个登录界面，通过账号位数来判断用户类型，以此避免了管理员特殊登录入口的存在。对于密码，采用 MD5 加密存储，且加密过程会将账号作为 salt，进一步提高安全性。

对于整个系统，用户有以下三类：学生（助教）、老师、教务（管理员），三类用户拥有不同的权限，后端通过账号的不同将主体分开，从而返回不同的主页。三类用户不共用前端页面，而是分别拥有各自能够进入的页面，从而可以很简单地限制三类用户不同的权限，只要在相应的主页提供相应的功能入口即可。

下面介绍每类用户被分配到的读写权限，同时简单介绍系统实现的功能。



## **1. 学生（助教）用户的读权限：**

学生可以查看已选课程、可选课程，对于可选课程，可以查看评价，对于已选课程，可以查看相关信息（如课程已选人数、自己的总分），还可以查看作业及反馈。此外，学生还可以查看自己在哪些课程里是助教身份。

对于拥有助教身份的课程，助教可以查看课程信息，选修课程的学生信息，每个学生完成作业的情况及提交作业详情。

## **2. 学生（助教）用户的写权限：**

学生可以选课、退课，可以提交/修改作业，可以提交免修、退学、转系申请，可以提交评教信息。

助教可以发布、修改、删除作业，批改作业，打平时分。

## **3. 老师用户的读权限：**

老师可以查看已开设课程及相关信息，对于每门课程，可以查看选修学生详情、助教相关信息、作业详情、免修申请，对于每次作业，可以查看学生提交详情。

## **4. 老师用户的写权限：**

老师可以加课、开课，对于每门课程，老师可以聘任、解聘助教，审批免修申请，发布、修改、删除作业，批改作业，打总分。

## **5. 教务用户的读权限：**

教务理论上可以查看任意信息。

## **6. 教务用户的写权限：**

教务理论上可以增加、更改、删除任意信息。

## 五、 存储过程、触发器和函数

如前文所述，在数据库实现过程中，项目直接调用了 django 提供的相关 API，通过自带 ORM 用 Python 相关类操作代替了直接的 SQL 语句。因此，在实际后端设计中，基本没有对数据库进行直接操作，故这三种功能也没有在实现中体现出来。不过，项目中运用 Python 函数等方式封装了类似的操作，虽然运行效率肯定不如 SQL 中原生的操作，但也让开发过程省了不少力。此外，为了演示对知识的掌握，数据库中仍然定义了存储过程、触发器、函数，但是没有实装，接下来将就此叙述。

### 1. 存储过程及函数：

存储过程是一组为了完成特定功能的 SQL 语句集合，可以返回多个值，也可以不返回任何值，而函数则是必定返回一个值（表项）的语句集合。利用存储过程和函数可以加速 SQL 语句的执行，且函数的针对性更强，两者的作用类似编程语言中“封装函数”的过程，可以将很多需要重复的过程简化。

在实现中，使用 Python 的函数定义实现类似存储过程与函数的操作。

在数据库中有如下存储过程定义：

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `getStudentSno`(IN Sno int)
BEGIN
    SELECT * FROM student WHERE Sno = Sno;
END
```

该存储过程可以通过学号查找相关学生的全部信息。

有如下函数定义：

```
CREATE DEFINER='root'@'localhost' FUNCTION `getTCs`(`TNO` int,`CNO` int,`DNO` int) RETURNS int
BEGIN
    DECLARE r int;
    SELECT s INTO r FROM dbmodels_tc WHERE tno_id = TNO AND dno_id = DNO AND Cno_id = CNO;
    RETURN r;
END
```

该函数可以通过课程相关信息查询该课程的选课人数。

## 2. 触发器：

触发器主要用于关联数据值的改变与相关表项的级联操作。对于前者，通常用于统计信息，在不考虑运行效率的情况下，可以用 SQL 函数（AVG, SUM 等）在每次使用时进行统计，也可以用 Python 函数进行类似实现；对于后者，主要用于外键的递归删除，而 MySQL 在创建外键时，只要将 on\_delete 设为 CASCADE，便可自动实现关联删除。

在实现中，除了上述替代之外，数据库中仍然实装了两个触发器，以统计选课人数。

下面的两个触发器将分别在有人选课（插入 SC）和退课（删除 SC）时改动相应 TC 表项的选课人数值：

```
CREATE TRIGGER tc_count_i AFTER INSERT ON dbmodels_sc FOR EACH ROW
UPDATE dbmodels_tc SET s = (SELECT count(*) FROM dbmodels_sc
                             WHERE TC_id = NEW.TC_id) WHERE id = NEW.TC_id;
```

```
CREATE TRIGGER tc_count_i AFTER DELETE ON dbmodels_sc FOR EACH ROW
UPDATE dbmodels_tc SET s = (SELECT count(*) FROM dbmodels_sc
                             WHERE TC_id = NEW.TC_id) WHERE id = NEW.TC_id;
```

## 六、 主要技术过程论述

项目使用 django 框架编写完成，运用 MTV 设计模式，具体实现分为表现层（用户层，前端）、数据层（后端，数据库）、逻辑层（视图层，后端，沟通用户端与数据库）

表现层（用户层）使用 HTML+CSS+JS 的组合，通过提交表单向逻辑层发

送 HttpRequest，并接受逻辑层返回的 HttpResponse，从而实现前后端通信。

逻辑层接受用户层发送的 HttpRequest，根据路由调用对应的后端处理程序，通过 HttpResponse 返回响应页面。django 的路由在 url 与视图之间建立映射关系，通过正则表达式的形式指定每个 url 对应的处理函数。如果处理过程设计数据库调用，则函数会通过 ORM 调用模型进行相关操作，否则直接返回一个网页。

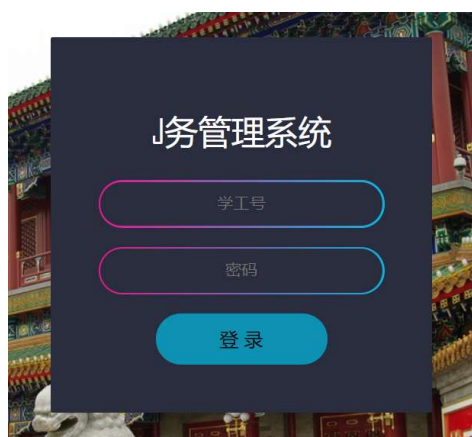
前端向后端发送数据通过表单传递 Request 实现，后端向前端返回数据通过字典形式，使用字典里的数据填充网页中的对应变量的。

数据层则使用 MySQL 作为数据库，通过 django 的 ORM 将对 MySQL 的直接操作映射为对 Python 对象的操作，实际对数据库的调用基本依赖 ORM。

在开发过程中，为了加快开发效率，使用 Python 函数实现了类似 SQL 存储过程、函数、触发器、视图的功能，实现了固定功能的封装，提高了项目的可读性、可维护性。

## 七、 系统功能的运行实例与功能逻辑

### 1. 登录界面



### 2. 学生用户

#### a) 选课相关

学生可以选择本专业的课程，同名课程只能选择一门，课程容量满后不能选择该门课程，选课时可以查看课程评价。

◆ 学生选课前（满容量不能再选）

教务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户：罗恩	注销
可选课程：								
课程名	授课老师	开课院系	学分	已选人数/容量	操作	查看评价		
魔咒课	邓布利多	格兰芬多	1	3/5	<a href="#">选课</a>	<a href="#">查看评价</a>		
面向麻瓜的魔法知识	邓布利多	格兰芬多	4	1/2	<a href="#">选课</a>	<a href="#">查看评价</a>		
保护神奇生物	斯内普	格兰芬多	3	2/10	<a href="#">选课</a>	<a href="#">查看评价</a>		
保护神奇生物	邓布利多	格兰芬多	3	0/10	<a href="#">选课</a>	<a href="#">查看评价</a>		
变形课	邓布利多	格兰芬多	4	1/2	<a href="#">选课</a>	<a href="#">查看评价</a>		
黑魔法防御术	麦格	格兰芬多	3	3/3	已满	<a href="#">查看评价</a>		
魔咒课	麦格	格兰芬多	1	0/5	<a href="#">选课</a>	<a href="#">查看评价</a>		
黑魔法防御术	邓布利多	格兰芬多	3	0/3	<a href="#">选课</a>	<a href="#">查看评价</a>		

◆ 课程评价

教务管理系统		已选课程		选课		评教		学籍异动				助教页面		当前用户：赫敏		注销													
概览		作业列表		评教																									
<div>课程名称：魔咒课</div> <div>授课教师：邓布利多</div> <div>开课院系：格兰芬多</div>																													
<div>课程评价：</div> <div>总体评价：4.0</div> <div>推荐率：50.0%</div> <div>专业性：3.5</div> <div>生动性：3.5</div>																													
<div>教师评价：</div> <div>总体评价：3.0</div> <div>专业性：2.75</div> <div>责任心：3.5</div>																													
<table><tr><th>课程评价</th><th>教师评价</th><th>是否推荐</th><th>评论</th></tr><tr><td>5</td><td>4</td><td>是</td><td>绝世好课！！</td></tr><tr><td>3</td><td>1</td><td>否</td><td>老师给分很吓人，慎选</td></tr></table>																		课程评价	教师评价	是否推荐	评论	5	4	是	绝世好课！！	3	1	否	老师给分很吓人，慎选
课程评价	教师评价	是否推荐	评论																										
5	4	是	绝世好课！！																										
3	1	否	老师给分很吓人，慎选																										

◆ 学生选课后（同名课程均不在列表中了）

[J务管理系统](#)
[已选课程](#)

127.0.0.1 显示  
选课成功！

[当前用户：罗恩](#)
[注销](#)

确定

可选课程：

课程名	授课老师	开课院系	学分	已选人数/容量	操作	查看评价
魔咒课	邓布利多	格兰芬多	1	3/5	<a href="#">选课</a>	<a href="#">查看评价</a>
面向麻瓜的魔法知识	邓布利多	格兰芬多	4	1/2	<a href="#">选课</a>	<a href="#">查看评价</a>
保护神奇生物	斯内普	格兰芬多	3	2/10	<a href="#">选课</a>	<a href="#">查看评价</a>
保护神奇生物	邓布利多	格兰芬多	3	0/10	<a href="#">选课</a>	<a href="#">查看评价</a>
变形课	邓布利多	格兰芬多	4	1/2	<a href="#">选课</a>	<a href="#">查看评价</a>
魔咒课	麦格	格兰芬多	1	0/5	<a href="#">选课</a>	<a href="#">查看评价</a>

### ◆ 学生已选课程

[J务管理系统](#)
[已选课程](#)
[选课](#)
[评教](#)
[学籍异动](#)
[助教页面](#)
[当前用户：弗雷德](#)
[注销](#)

已选课程：

课程名	授课老师	成绩	操作
保护神奇生物	斯内普	未结课	<a href="#">课程详情</a>
面向麻瓜的魔法知识	邓布利多	免修 80	<a href="#">课程详情</a>
魔咒课	邓布利多	未结课	<a href="#">课程详情</a>

### ◆ 课程详情页（结课后）

[J务管理系统](#)
[已选课程](#)
[选课](#)
[评教](#)
[学籍异动](#)
[助教页面](#)
[当前用户：弗雷德](#)
[注销](#)

[概览](#)
[作业列表](#)
[评教](#)

课程名称：面向麻瓜的魔法知识

授课教师：邓布利多

开课院系：格兰芬多

课程容量：1/2

学生名称：弗雷德

成绩：免修 80

## b) 作业及免修相关

学生可以多次提交已发布的作业，提交次数达上限后不能再次提交，在评分之后，学生可以查看反馈。学生可以对课程提交免修申请，每门课程只能提交一次。若审批通过，则课程结课。

## ◆ 提交免修申请

127.0.0.1 显示  
免修申请已提交!

确定

J务管理系统

已选课程

选课

助教页面

当前用户: 赫敏

注销

概览

作业列表

评教

授课教师: 邓布利多

开课院系: 格兰芬多

课程容量: 3/5

学生名称: 赫敏

成绩: 未结课

申请理由:

申请免修

退课

## ◆ 作业列表

J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户: 弗雷德	注销
概览	作业列表	评教						
作业标题		提交次数	作业得分	操作				
阿瓦达索命		1/ 3	2.0	查看作业				
自由课题		0/ ∞	未提交	查看作业				
魔咒研发		0/ 4	未提交	查看作业				

## ◆ 提交作业

J务管理系统

已选课程

选课

评教

学籍异动

助教页面

当前用户: 弗雷德

注销

概览

作业列表

评教

作业标题: 自由课题

提交次数上限: 无限制

已提交次数: 0

得分: 未提交

作业内容:

挑选一个咒语, 对其进行调研

提交内容:

XXXXXX

提交作业

## ◆ 查看作业反馈

J务管理系统		已选课程	选课	评教	学籍异动			助教页面	当前用户：弗雷德	注销
概览	作业列表	评教								
<div>作业标题：阿瓦达索命</div> <div>提交次数上限：3</div> <div>已提交次数：1</div> <div>得分：2.0</div> <div>作业内容：</div> <div>练习使用阿瓦达索命</div>										
<div>得分：2.0</div> <div>评阅反馈：完成态度不佳</div>										

## c) 评教相关

学生可以在结课之后对课程进行评教，评教结果可以供教务与其他学生参考。

### ◆ 评教首页

J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户: 乔治	注销
--------	------	----	----	------	--	------	----------	----

课程名	授课老师	成绩	评教
魔咒课	邓布利多	免修 95	已评教
保护神奇生物	斯内普	未结课	未结课
变形课	邓布利多	免修 86	已评教
黑魔法防御术	麦格	未结课	未结课

### ◆ 评教页



J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户: 弗雷德	注销
概览	作业列表	评教						

课程名称: 魔咒课

授课教师: 邓布利多

开课院系: 格兰芬多

课程容量: 2/5

学生名称: 弗雷德

成绩: 未结课

课程总体评价: ☐0 ☐1 ☐2 ☐3 ☒4 ☐5

课程专业程度: ☐0 ☐1 ☐2 ☒3 ☐4 ☐5

课程生动程度: ☐0 ☐1 ☐2 ☐3 ☐4 ☒5

---

老师总体评价: ☐0 ☐1 ☐2 ☐3 ☒4 ☐5

老师专业程度: ☐0 ☐1 ☐2 ☐3 ☐4 ☒5

老师负责程度: ☐0 ☐1 ☐2 ☐3 ☐4 ☒5

---

是否推荐课程: ☒是 ☐否

课程评价:

这课很好, 推荐!

未结课

#### d) 学籍相关

学生可以申请转系、退学, 这两种申请在没有审批完成前都不能进行下一次申请。

##### ◆ 退学申请

J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户: 弗雷德	注销
--------	------	----	----	------	--	------	-----------	----

学生名称: 弗雷德

学生学号: 13131313

所属院系: 格兰芬多

---

申请理由:

我不想学了!!

提交退学申请

教务同意申请后, 请自行再次登录教务系统确认, 之后将删除账号!

##### ◆ 转系申请

J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户: 哈利	注销
--------	------	----	----	------	--	------	----------	----

学生名称: 哈利

学生学号: 11111111

当前院系: 格兰芬多

---

目标院系:

申请理由:

我想学CS

提交转系申请

须知: 请在已选的本专业课程全部结课之后再申请转系!

### 3. 助教/教师用户

#### a) 助教用户

助教是由老师聘任/解聘的，老师只能将已经完成课业的学生设为助教。助教页面与教师页面类似，不过一级导航栏为学生栏。

##### ◆ 设置助教

教务管理系统		已开课程		开设/增加课程		当前用户：邓布利多		注销	
概览	发布作业	作业列表	打分	免修审批					
课程名称：魔咒课									
授课教师：邓布利多									
开课院系：格兰芬多									
课程容量：3/5									
姓名		成绩			助教聘任				
乔治		免修 95			解任助教				
弗雷德		55			设为助教				
赫敏		未结课			学习中				

##### ◆ 学生助教入口

J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户：乔治	注销
课程名	开课院系	授课教师	学分	选课人数/容量		操作		
魔咒课	格兰芬多	邓布利多	1	2/5		进入管理页面		

#### b) 课程相关（教师）

老师可以增加课程，并在特定院系开设课程，还可以审批免修申请。

##### ◆ 课程操作

J务管理系统

已开课程

开设/增加课程

当前用户: 邓布利多

注销

开设课程

课程名字

魔咒课

开课院系

拉文克劳

开课

增加课程

课程名字

请输入课程名

课程学分

请输入课程学分

课程容量

请输入课程容量

增加课程

◆ 课程首页（助教，与老师端基本相同）

J务管理系统

已选课程

选课

评教

学籍异动

助教页面

当前用户: 乔治

注销

概览

发布作业

作业列表

打平时分

课程名称: 魔咒课

授课教师: 邓布利多

开课院系: 格兰芬多

课程容量: 2/5

姓名	成绩
乔治	免修 95
弗雷德	未结课

◆ 免修申请列表

J务管理系统

已开课程

开设/增加课程

当前用户: 邓布利多

注销

概览

发布作业

作业列表

打分

免修审批

姓名	院系	操作
乔治	格兰芬多	已同意
弗雷德	格兰芬多	已拒绝
赫敏	格兰芬多	处理

◆ 免修审批

教务管理系统	已开课程	开设/增加课程			当前用户: 邓布利多	注销
概览	发布作业	作业列表	打分	免修审批		

学生姓名: 赫敏

所属院系: 格兰芬多

---

申请理由: 我都会了

处理意见

☒ 同意 ☐ 拒绝

免修得分:

## c) 作业相关

老师/助教均可以布置作业、批改作业、修改作业信息和删除作业。

### ◆ 发布作业 (老师)

教务管理系统	已开课程	开设/增加课程			当前用户: 邓布利多	注销
概览	发布作业	作业列表	打分	免修审批		

作业标题:

作业内容:

☒ 限制提交次数

限制次数:

### ◆ 作业列表 (老师)

教务管理系统	已开课程	开设/增加课程			当前用户: 邓布利多	注销
概览	发布作业	作业列表	打分	免修审批		

作业标题	提交上限	提交人数	操作
阿瓦达索命	3	1	<a href="#">作业详情</a>
自由课题	无上限	0	<a href="#">作业详情</a>
魔咒研发	4	0	<a href="#">作业详情</a>

### ◆ 作业详情 (助教)

J务管理系统		已选课程	选课	评教	学籍异动		助教页面	当前用户: 乔治	注销
概览	发布作业	作业列表	打平时分						

作业标题: 阿瓦达索命

提交次数上限: 3

已提交人数: 1

作业内容:

练习使用阿瓦达索命

[删除作业](#)

[修改作业信息](#)

学生姓名	提交次数	作业成绩	作业评阅
乔治	0	未提交	尚未提交
弗雷德	1	未评阅	<a href="#">评阅作业</a>

◆ 批改作业（助教）

J务管理系统		已选课程	选课	评教	学籍异动		助教页面	当前用户: 乔治	注销
概览	发布作业	作业列表	打平时分						

作业标题: 阿瓦达索命

提交次数上限: 3

已提交次数: 1

得分: 未评阅

作业内容:

练习使用阿瓦达索命

---

作业评阅:

学生提交作业:

我不会

分数:

2

完成态度不佳

[评阅](#)

d) 打分相关

助教可以给学生打平时分，在助教打了平时分之后，老师可以打总分。在打分时，可以看到学生平时表现。

◆ 打分列表（助教）

J务管理系统	已选课程	选课	评教	学籍异动		助教页面	当前用户: 乔治	注销
概览	发布作业	作业列表	打平时分					
姓名	成绩	作业提交情况	已批阅作业平均分	平时分				
乔治	免修 95	0/2	0	已结课				
弗雷德	未结课	1/2	2.0	<a href="#">打分</a>				

◆ 打分列表（老师，打分完成后的表现）

J务管理系统

已开课程

开设/增

概览

发布作业

作业列表

打分

免修审批

127.0.0.1 显示  
打分完成!

确定

姓名	作业提交情况	已批阅作业平均分	平时分	成绩
乔治	0/2	0	已结课	免修 95
弗雷德	1/2	2.0	已结课	55
赫敏	0/2	0	未打分	等待平时分

◆ 打平时分

J务管理系统

已选课程

选课

评教

学籍异动

助教页面

当前用户: 乔治

注销

概览

发布作业

作业列表

打平时分

学生姓名: 弗雷德

历次作业得分:

作业标题: 阿瓦达索命

作业得分: 2.0

作业标题: 自由课题

作业得分: 未提交

平时分:  [打分](#)

◆ 打总分

J务管理系统		已开课程	开设/增加课程				当前用户: 邓布利多	注销
概览	发布作业	作业列表	打分	免修审批				

学生姓名: 弗雷德

历次作业得分:

---

作业标题: 阿瓦达索命

作业得分: 2.0

---

作业标题: 自由课题

作业得分: 未提交

---

平时分: 45

总分:

## 4. 教务用户

### a) 统计报表与用户操作

教务可以录入学生、老师，增加院系，还可以看到各种统计报表信息。

#### ◆ 学生管理

学生管理	教师管理	院系管理	课程管理	事务审批	J务管理系统-admin	当前用户: admin	注销
------	------	------	------	------	--------------	-------------	----

学号

请输入学生学号

姓名

请输入学生姓名

系

拉文克劳 ▾

学生信息:

学生名称	所属院系	选课数量	已修得学分	加权平均分
哈利	计算机学院	0/0	0	0
乔治	格兰芬多	2/4	5	87.8
弗雷德	格兰芬多	2/4	5	75.0
赫敏	格兰芬多	1/2	1	99.0
罗恩	格兰芬多	0/0	0	0
伍德	格兰芬多	0/0	0	0

#### ◆ 教师管理

学生管理	教师管理	院系管理	课程管理	事务审批	J务管理系统-admin	当前用户: admin	注销
------	------	------	------	------	--------------	-------------	----

工号:  
请输入教师工号

姓名:  
请输入教师姓名

录用教师

教师信息:

教师名称	开课数量	总体评价
邓布利多	8	3.0
麦格	4	0
斯内普	3	0
伏地魔	0	0

## ◆ 院系管理

学生管理	教师管理	院系管理	课程管理	事务审批	J务管理系统-admin	当前用户: admin	注销
------	------	------	------	------	--------------	-------------	----

系号:  
请输入院系系号

系名:  
请输入院系系名

增加院系

## ◆ 课程信息

学生管理	教师管理	院系管理	课程管理	事务审批	J务管理系统-admin	当前用户: admin	注销
课程信息:							
课程名称	所属院系	授课老师	选修人数	平均分	挂科率	良好率	优秀率
魔咒课	格兰芬多	邓布利多	3/3	83.0	33.33333333333333%	0.0%	66.66666666666666%
魔药学	计算机学院	邓布利多	0/0	0	0%	0%	0%
黑魔法防御术	计算机学院	邓布利多	0/0	0	0%	0%	0%
面向麻瓜的魔法知识	格兰芬多	邓布利多	1/1	80.0	0.0%	0.0%	100.0%
保护神奇生物	斯莱特林	邓布利多	0/0	0	0%	0%	0%
黑魔法防御术	计算机学院	麦格	0/0	0	0%	0%	0%
保护神奇生物	斯莱特林	斯内普	0/0	0	0%	0%	0%
保护神奇生物	格兰芬多	斯内普	0/2	0	0%	0%	0%
黑魔法防御术	斯莱特林	斯内普	0/0	0	0%	0%	0%

## b) 学籍异动审批

教务可以审批学生的学籍异动申请。



◆ 审批首页

学生管理

教师管理

院系管理

课程管理

事务审批

J务管理系统-admin

当前用户: admin

注销

转系申请

姓名	当前院系	目标院系	操作
哈利	计算机学院	计算机学院	已同意
纳威	格兰芬多	赫奇帕奇	处理

退学申请

姓名	院系	操作
弗雷德	格兰芬多	已拒绝
纳威	格兰芬多	处理

◆ 退学审批

学生管理

教师管理

院系管理

课程管理

事务审批

J务管理系统-admin

当前用户: admin

注销

学生姓名: 纳威

学生学号: 44444444

所属院系: 格兰芬多

申请理由: 我要回家种田

处理意见

☒ 同意 ☐ 拒绝

处理

◆ 转系审批

学生管理

教师管理

院系管理

课程管理

事务审批

J务管理系统-admin

当前用户: admin

注销

学生姓名: 哈利

学生学号: 11111111

当前院系: 格兰芬多

目标院系: 计算机学院

申请理由: 我想学CS

处理意见

☒ 同意 ☐ 拒绝

处理

八、 源程序简要说明

工程使用 MTV 设计模式，故源程序也主要建立在这三个层面的基础上。

1. 源程序的主要框架

a) 模型

对应工程目录下的 DBmodels 文件夹，其中 models.py 定义了数据类与

其相关约束，其中每个类对应了 MySQL 中的一个表关系，而其中的约束大多不是建立在 MySQL 基础上，而是在调用 ORM 语句时进行的判断。

## **b) 模板**

对应工程目录下的 templates 文件夹，其中的每个 HTML 文件都对应着一个页面，页面中有需要动态生成的变量，本工程全部采用 HttpResponse 的形式返回字典，用字典里的同名变量进行填充。

## **c) 视图**

对应工程目录下的工程名文件夹，其中的 py 文件定义控制函数，说明对于相应请求需要做出的相应响应。django 通过 urls.py 里定义的路由分配映射关系，指定特定 url 对应的功能函数。在进入控制函数后，函数通过表单发送的 request 接受参数，进行操作后通过 render 函数返回特定页面。在本工程中，一切约束均在后端进行，如果需要报错或警告，也均由控制函数返回相应的提示语句，并在不正常的调用下不执行相关操作。

# **2. 控制函数详述**

在本工程的实现中，每个控制函数都对应着 render 返回语句，也即每个操作都会重新进入一次页面（即便是原页面），故以下控制函数的类别主要针对过程操作。控制函数基本是以下两种功能的复合，部分函数只有查询和跳转功能。

## **1) 查询数据**

每个页面中都会有相应的动态展示内容（如：学生选课界面需要知道可选课程的相关信息），故每个控制函数在返回之前都会先对字典进行相应填充，而为了方便起见，后端将页面分为了学生、老师、学生课程页、老师课程页、助

教课程页、教务六种类型，在进入每种页面前，统一查询该页面~~可能~~用到的所有信息，统一查询过程封装在 `ctxf.py` 中，查询需要用到对应数据类（如：学生课程页需要将 SC 表项作为参数）。

## 2) 更改数据

读取参数，调用 ORM 对数据库进行操作。需要注意的是，因为 ORM 没法方便地在数据库操作异常时返回错误类型，故约束判断均以后端手动编写的方式进行。如：录入学生时首先查表，若学号重复则不执行 `create`（对应 SQL 语句中的 `INSERT`）操作，直接返回提示信息。

# 九、 小组分工与心得体会

## 1. 小组分工

**吕云帆**：负责 `django` 前后端实现，项目总设计，撰写设计与实现报告，制作 PPT 并进行申优展示。

**马瑞**：负责前端 `CSS+JS` 部分，服务器部署相关，`git` 仓库部署与管理，项目监督，参与项目设计。

**马逸行**：负责前端 `CSS` 部分，负责项目初版设计，参与后续设计讨论与改动。

## 2. 心得体会

这次数据库大作业是一个综合性很强的项目，我们不仅需要掌握数据库相关知识与应用方式，更要学习如何实现一个完整的应用，这对于我们来说可谓是一个全新的领域，一切都要从头学习，但这也同样代表着这注定是一次收获颇丰的旅程。

在这次课程设计中，我们最大的收获便是知道了“规划”的重要性，这不仅

体现在方案设计上，同样也适用于分工与工期分配上。对于项目来说，前期的项目设计十分重要，若是在一开始就定下了完善的计划与设计，并将任务拆分为适合分离实现的多个部分，同时制定张弛有度的工期计划，则每个人便可以各司其职，有条不紊地进行开发，工程也可以不紧不慢地完成，最后还会剩出许多时间进行改进、完善和迭代开发。如果初期规划不到位，则之后便会频出差错：首先是必须不断改变设计，继而改变前后端的相应设计，而这很不利于多人合作的项目；其次是分工上，分工的不合理可能会导致小组合作失去其原有的意义，也会影响开发进度；此外还有工期计划，即使有宏图在胸，如果不着手去实现，最后也只能手忙脚乱地进行开发，无法展现出应有的水平。

因为统筹上的失误，我们几乎遇到了上述每个问题，但我们最后还是齐心协力，一起完成了这个项目，而且结果在主观上还算满意。在这次项目中，我们不仅巩固了课上学习的数据库相关知识，还在繁杂的开发过程体会到了 SQL 语言的妙处，还感受到了亲手从零开始构建项目的成就感。此外，我们还学习到了许多其他知识，如 django、CSS、JS 等等。而且，我们还切身体会到了合作中每个人的不可或缺，如果没有马瑞负责 git 与服务器的部署管理，我们便不可能多线同步开发、实时测试，项目开发的进度将拖慢很多；如果没有马逸行在前端上的努力，项目最终的效果将只有简陋的 HTML 页面，即使后端再出色也无法发出应有的光彩；如果没有吕云帆在工期延误后“挑起大梁”，整个项目则很难有这样的完成度。