

**在线考试系统**

项目组成员: 江哲倩、陈树斌、陈耿钊、黎浩贤

所在班级: 17级计算机A班

[1. 获取需求 1](#_Toc30752)

[1.1需求提出 1](#_Toc13756)

[1.2功能需求分析 1](#_Toc16243)

[1.3 非功能需求分析 2](#_Toc5401)

[2. 系统设计 2](#_Toc19571)

[2.1数据库设计 2](#_Toc2934)

[2.2用例建模 5](#_Toc13677)

[2.3类图建模 10](#_Toc32758)

[2.4行为建模 12](#_Toc3776)

[3.系统整体架构设计 16](#_Toc33)

[4.代码实现 17](#_Toc22513)

[5.测试过程 27](#_Toc32027)

[5.1链接测试 27](#_Toc13946)

[5.2表单测试 27](#_Toc17336)

[5.3内容测试 27](#_Toc5149)

[5.4数据库测试 28](#_Toc3588)

[6.验证需求 28](#_Toc9404)

[6.1一致性需求 28](#_Toc16953)

[6.2 完整性需求 28](#_Toc14343)

[6.3现实性需求 28](#_Toc21050)

[7.总结 29](#_Toc7551)

# 1. 获取需求

1.1需求提出

目前由于移动互联网的飞速发展和普及，为如今的教育事业带来了新的改革和挑战。我国教育行业带来新的机遇，互联网+教育的教学新形式逐渐出现在大众的视野，相比较下，传统的教学存在着一定的局限。比如传统的考试形式大多采用纸质考试，按照人工出题、审核、成绩评估、统计结果的流程进行，更甚至在落后贫困的山区的学生的考卷甚至是手抄的。考试是教学过程中的一个重要环节，纸质考试的方式往往使人们受到时间和空间等各种各样的限制，造成很多不利。传统的纸质考试不仅需要花费大量的纸质资源，而且增加教师的工作负担，使教师得不到及时的教学反馈。

1.2功能需求分析

在线考试系统要求实现网络考试的各项基本功能，旨在解决传统的纸质考试中繁琐的人工操作过程以及教师对学生成绩的统计过程，更加直观、高效获取有价值的信息。本系统功能主要有系统的登录退出、在线考试、成绩查询、用户管理、试题信息管理、组卷管理、成绩统计等功能。

1. 系统登录

有系统后台创建超级管理员，管理员用预先提供的账号密码登录系统；普通用户进入系统前由管理员提前录入的账号密码进行验证登录，根据用户的不同身份进入不同的页面。

1. 在线考试：

考生使用时，根据学生的班级和登录时间对应显示应考科目的学生学号、姓名、考试科目、考试时间，点击开始考试进入当前科目的考试，显示当前考试的试卷内容。当学生完成试卷的提交或考试时间结束，系统不再允许学生进行作答或者修改试卷；系统提供考试时间倒计时的功能，方便学生随时了解考试剩余时间；系统将学生提交的试卷实现自动评阅，并记录学生的考试成绩，将评阅结果提供给学生。

（3）成绩查询：供教师和学生使用，分为学生查询成绩和教师查询成绩，为不同用户提供不同查询方式。成绩的录入是系统根据交卷进行自动评分来进行自动录入汇总，教师和学生登录系统后，通过对应的模块进行成绩查询。教师还可以按班级进行考试成绩的查询。

1. 用户管理：超级管理员是系统权限操作者，可以对系统管理员、学生、教师用户的基本信息进行增删改查以及维护。

（5）试题管理：供教师和管理员使用，可以在题库中对试题进行添加、修改操作，维护题库。

（6）组卷：供教师和管理员使用，可以根据考试科目，从题库中选择一些符合条件的试题，形成一份试卷。为了方便组卷，应提供方便的查询功能，使教师能查询不同要求的试题，还可以对试卷题进行添加、修改操作。

（7）成绩统计：供教师和管理员使用，按照科目、班级等统计学生的考试成绩。

1.3 非功能需求分析

　　基于B/S架构的在线考试系统通过浏览器（Browser）和服务器（Server）交互的方式实现，在登录、管理、考试这些整体功能的界面应当遵循界面简洁、操作简单的原则，避免繁琐问题给用户增加工作量等，具体的要求包括：

1.3.1 界面美观

　　界面是进行人加交互的接口，根据大部分的Web应用的操作方式进行UI的布局设计，具备良好的UI界面是吸引用户使用的最直接因素。因此系统的操作界面整体采用暗淡的黄色背景作为导航栏的显示，黑色的背景作为主要功能的显示。

1.3.2 操作简单

　　系统的操作需要同时兼顾管理者和考生的使用需求，同时更要对常用功能和易出错功能给与人性化的提示，并且能够拥有操作结果的反馈信息。

1.3.3 多用户在线考核

　　本系统是目标用户是学生群体，系统至少需要能够60个考生同时进行在线考试的并发量要求。

1.3.4 运行稳定

　　在线考试系统是教育考核的一种新方式，在考核的过程中必须保证应用的稳定性，避免发生崩溃导致考生需要重复考等故障。

# 2. 系统设计

2.1数据库设计

在线考试系统数据库分为大模块：学生信息表、教师信息表、课程信息表、试题信息表、作答记录表，具体描述如下

（1）学生信息表

表 1 学生信息表（student）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段类型 | 字段长度 | 备注 |
| sid | int | 11 | 唯一主键，no null |
| name | varchar | 20 | no null |
| sex | varchar | 4 | no null |
| dept | varchar | 20 | no null |
| major | varchar | 20 | no null |
| password | varchar | 20 | no null |
| email | varchar | 254 | no null |
| birth | date | 20 | no null |

（2）教师信息表

表 2 教师信息表（teacher）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段类型 | 字段长度 | 备注 |
| tid | int | 11 | 唯一主键，no null |
| name | varchar | 20 | no null |
| sex | varchar | 4 | no null |
| dept | varchar | 20 | no null |
| password | varchar | 20 | no null |
| email | varchar | 254 | no null |
| birth | date | 20 | no null |

（3）题库信息表

表 3 题库信息表（question)

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段类型 | 字段长度 | 备注 |
| qid | int | 11 | 唯一主键,no null |
| subject | varchar | 20 | no null |
| title | longtext | 0 | no null |
| optionA | varchar | 30 | no null |
| optionB | varchar | 30 | no null |
| optionC | varchar | 30 | no null |
| optionD | varchar | 30 | no null |
| answer | varchar | 10 | no null |
| level | varchar | 10 | no null |
| score | int | 11 | no null |

1. 考试信息表

表 4 考试信息表（paper）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段类型 | 字段长度 | 备注 |
| testid | int | 11 | 唯一主键，no null |
| subject | varchar | 20 | no null |
| major | varchar | 20 | no null |
| examtime | datetime | 6 | no null |
| pid | int | 11 | no null |
| tid | int | 11 | no null |

1. 成绩信息表

表 5 成绩信息表（grade）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段类型 | 字段长度 | 备注 |
| sid | int | 11 | 唯一主键，no null |
| subject | varchar | 20 | no null |
| sname | varchar | 20 | no null |
| grade | int | 11 | no null |

1. 试卷信息表

表 6 试卷信息表（paper）

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 字段类型 | 字段长度 | 备注 |
| pid | int | 11 | 唯一主键，no null |
| testid | int | 11 | no null |
| qid | int | 11 | no null |

2.2用例建模



图 1学生用例图



图 2 管理员用例图



图 3 教师用例图

2.2.1添加题库题目

表格 1 添加题库题目

|  |  |
| --- | --- |
| 用例 | 添加题库题目 |
| 简单描述 | 这个用例允许管理员、教师登录系统，进入在线考试系统后台进行题库的的增加操作。 |
| 角色（参与者） | 教师、管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击单项选择题题库按钮请求增加题库信息 3. 在线考试系统后台显示增加题库信息页面 4. 依次输入需要考试的科目名称、题目内容、题目选项、题目答案，并点击保存添加添加 5. 系统保存题库题目的添加操作，并更新题库信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行添加，系统显示出错页面，要求重新输入所缺信息 2. 若添加的题目重复，系统提示已存在题目，请勿重复添加 3. 系统出现故障，提示添加题库失败 |
| 后置条件 | 如果用例成功，添加题库的相关信息并记录到数据库。否则，系统状态不变 |

2.2.2修改题库题目

表格 2 修改题库题目

|  |  |
| --- | --- |
| 用例 | 修改题库题目 |
| 简单描述 | 这个用例允许管理员、教师登录系统，进入在线考试系统后台进行题库的的修改操作。 |
| 角色（参与者） | 教师、管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击单项选择题题库按钮请求修改题库信息 3. 在线考试系统后台显示修改题库信息页面 4. 查看选择对应的ID、科目、题目、以及选项、答案进行修改，进行内容的修改操作，保存 5. 系统保存题库信息的修改操作，并更新题库信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行添加，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示添加题库失败 |
| 后置条件 | 如果用例成功，添加题库的相关信息并记录到数据库。否则，系统状态不变 |

2.2.3 添加考试试卷

表格 3 添加考试试卷

|  |  |
| --- | --- |
| 用例 | 添加考试试卷 |
| 简单描述 | 这个用例允许教师、管理员登录系统，进入在线考试系统后台进行试卷的增加操作。 |
| 角色（参与者） | 教师、管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击试卷按钮请求增加考试试卷功能 3. 在线考试系统后台显示增加试卷信息页面 4. 依次输入需要考试的科目名称、内容、ID、考试日期、考试时间，并点击保存添加添加 5. 系统保存考试试卷的添加操作，并更新考试信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行添加，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示添加考试科目失败 |
| 后置条件 | 如果用例成功，添加考试试卷的相关信息并记录到数据库。否则，系统状态不变 |

2.2.4修改考试试卷

表格 4 修改考试试卷

|  |  |
| --- | --- |
| 用例 | 修改考试试卷 |
| 简单描述 | 这个用例允许教师、管理员登录系统，进入在线考试系统后台进行试卷的修改操作。 |
| 角色（参与者） | 教师、管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击试卷按钮请求修改考试试卷功能 3. 在线考试系统后台显示修改试卷信息页面 4. 查看对应的任课教师、考试科目以及考试日期时间的试卷进行选择，并点击修改，进行内容的修改操作 5. 系统保存考试试卷的修改操作，并更新考试内容 |
| 可选事件流 | 1. 未提供系统所要求操作进行修改，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示修改考试科目失败 |
| 后置条件 | 如果用例成功，修改考试试卷的相关信息并记录到数据库。否则，系统状态不变 |

2.2.5添加用户

表格 5 添加用户

|  |  |
| --- | --- |
| 用例 | 添加用户 |
| 简单描述 | 这个用例允许管理员登录，进入在线考试系统后台进行用户的添加操作 |
| 角色（参与者） | 管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击学生或教师按钮请求增加用户的相关信息 3. 在线考试系统后台显示增加用户信息页面 4. 依次输入学生的学号、姓名、性别、学院、专业、密码、邮箱、出生日期（教师的教工号、姓名、性别、学院、邮箱、密码、出生日期）并点击保存添加 5. 系统保存题库信息的修改操作，并更新题库信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行添加，系统显示出错页面，要求重新输入所缺信息； 2. 若添加的用户重复，系统提示已存在，请勿重复添加 3. 系统出现故障，提示添加用户失败 |
| 后置条件 | 如果用例成功，添加用户的相关信息并记录到数据库。否则，系统状态不变 |

2.2.6修改用户信息

表格 6 修改用户信息

|  |  |
| --- | --- |
| 用例 | 修改用户信息 |
| 简单描述 | 这个用例允许管理员登录系统，进入在线考试系统后台进行用户的修改操作。 |
| 角色（参与者） | 管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击学生或教师请求修改用户信息 3. 在线考试系统后台显示修改用户信息页面 4. 查看选择对应的学生的学号、姓名、性别、学院、专业、密码、邮箱、出生日期（教师的教工号、姓名、性别、学院、邮箱、密码、出生日期）进行修改，进行内容的修改操作，保存 5. 系统保存题库信息的修改操作，并更新用户信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行修改，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示修改与信息失败 |
| 后置条件 | 如果用例成功，修改用户的相关信息并记录到数据库。否则，系统状态不变 |

2.2.7添加成绩信息

表格 7 添加成绩信息

|  |  |
| --- | --- |
| 用例 | 添加成绩信息 |
| 简单描述 | 这个用例允许管理员登录，进入在线考试系统后台进行成绩的添加操作 |
| 角色（参与者） | 管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击成绩钮请求增加用户的相关信息 3. 在线考试系统后台显示增加成绩信息页面 4. 选择学生学号、科目依次输入学生的成绩，并点击保存添加 5. 系统保存成绩信息的增加操作，并更新成绩信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行添加，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示添加成绩失败 |
| 后置条件 | 如果用例成功，添加成绩的相关信息并记录到数据库。否则，系统状态不变 |

2.2.8修改成绩信息

表格 8 修改成绩信息

|  |  |
| --- | --- |
| 用例 | 修改成绩信息 |
| 简单描述 | 这个用例允许管理员登录系统，进入在线考试系统后台进行成绩的修改操作。 |
| 角色（参与者） | 管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统后台 |
| 主事件流 | 1. 进入在线考试系统后台的站点管理 2. 点击成绩请求修改用户信息 3. 在线考试系统后台显示修改成绩信息页面 4. 查看选择对应的学生的学号、科目进行修改，进行成绩的修改操作，保存 5. 系统保存成绩信息的修改操作，并更新成绩信息列表 |
| 可选事件流 | 1. 未提供系统所要求操作进行修改，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示修改成绩信息失败 |
| 后置条件 | 如果用例成功，修改成绩的相关信息并记录到数据库。否则，系统状态不变 |

2.2.9在线考试

表格 9 在线考试

|  |  |
| --- | --- |
| 用例 | 在线考试信息 |
| 简单描述 | 这个用例允许学生用户登录，进入在线考试系统进行在线考试 |
| 角色（参与者） | 学生 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统 |
| 主事件流 | 1. 进入在线考试系统学生界面，点击考试信息 2. 在线考试系统显示考试信息页面，显示考试的学生学号、姓名、考试科目、考试时间，考卷内容 3. 学生进行作答以及保存、提交 4. 系统保存学生作答提交记录，评定成绩信息给学生，并保存数据库中 |
| 可选事件流 | 1. 未提供系统所要求操作进行提交，系统显示出错页面，要求学生重新提交所缺信息； 2. 系统出现故障，提示提交试卷失败 |
| 后置条件 | 如果用例成功，提交试卷的相关信息并记录到数据库。否则，系统状态不变 |

2.2.10成绩查询

表格 10 成绩查询

|  |  |
| --- | --- |
| 用例 | 成绩查询 |
| 简单描述 | 这个用例允许学生、教师、管理员登录，进入在线考试系统进行成绩查询操作 |
| 角色（参与者） | 学生、教师、管理员 |
| 前置条件 | 通过登录账号密码登录进入在线考试系统 |
| 主事件流 | 1. 进入在线考试系统用户界面，点击成绩查询 2. 可根据输入学生学号继续查询 3. 系统显示该学生姓名、考试科目、考试成绩的相关信息 |
| 可选事件流 | 1. 未提供系统所要求操作进行查询，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示查询成绩失败 |
| 后置条件 | 如果用例成功，系统将显示成绩的相关信息。否则，系统状态不变 |

2.2.11成绩统计

表格 11 成绩统计

|  |  |
| --- | --- |
| 用例 | 成绩统计 |
| 简单描述 | 这个用例允许教师登录系统，进行学生成绩的统计 |
| 角色（参与者） | 教师 |
| 前置条件 | 教师通过登录账号密码登录进入在线考试系统 |
| 主事件流 | 1. 进入在线考试系统教师用户界面，点击成绩统计 2. 选择对应的任课教师、考试科目、考试时间，查看学生成绩 3. 选择不同统计图，系统将进行成绩统计； 4. 系统显示该科目的成绩统计情况 |
| 可选事件流 | 1. 未提供系统所要求操作进行成绩统计，系统显示出错页面，要求重新输入所缺信息； 2. 系统出现故障，提示失败 |
| 后置条件 | 如果用例成功，系统将统计结果的相关信息显示记录并更新考试信息到数据库。否则，系统状态不变 |

2.3类图建模



图 4 学生类 图 5 管理员类 图 6 教师类



图 7 题目类 图 8成绩类 图 9考试类 图 10试卷类



图 11 系统类图

2.4行为建模

2.4.1在线考试状态图



图 12 在线考试状态图

2.4.2查询成绩状态图



图 13 查询成绩状态图

2.4.3 添加题目/试卷状态图



图 14 添加题目/试卷状态图

2.2.4 添加用户状态图



图 15 添加用户状态图

2.2.5 在线考试顺序图



图 16 在线考试顺序图

2.2.6添加用户顺序图



图 17 添加用户顺序图

2.2.7 添加试卷顺序图



图 18 添加试卷顺序图

2.2.8 修改试卷顺序图



图 19 修改试卷顺序图

2.2.9 查询成绩顺序图



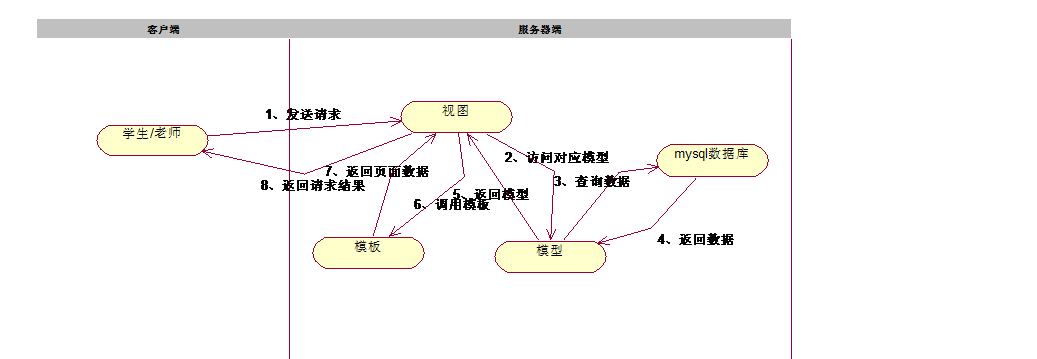
图 20 查询成绩顺序图

# 3.系统整体架构设计

Django是基于MVC构造的框架，在本系统中，控制器接受用户输入的部分由框架自行处理，所以此更关注的是模型（M）、模板（T）、和视图（V），也可以说是MVT模式。

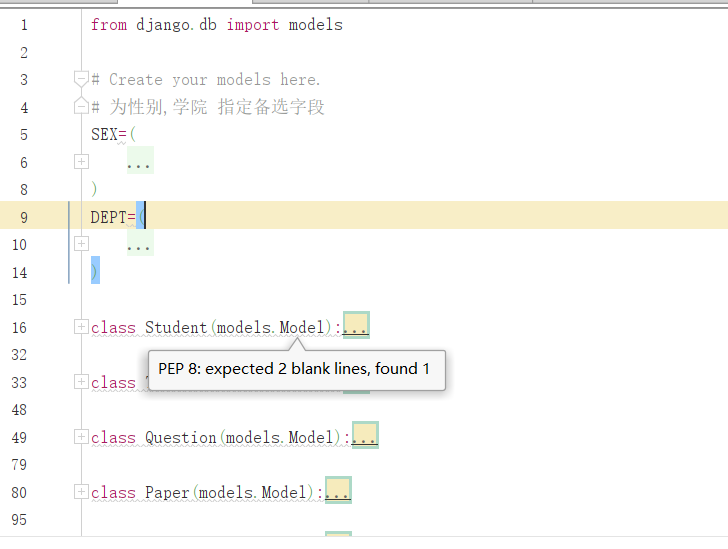
MVT各自的职责分别是：

|  |  |
| --- | --- |
| 层次 | 职责 |
| 模型（Model），即数据存取层 | 数据存储 |
| 模板（Template），即表现层 | 处理与表现相关的决定 |
| 视图（View），即业务逻辑层 | 存取模型及调取恰当模板的相关逻辑 |

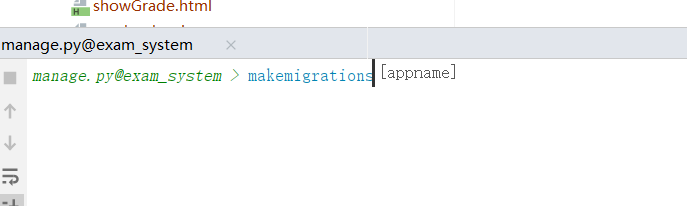


# 4.代码实现

1. 建表



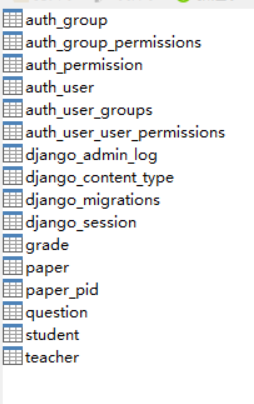
1. 将模型映射到mysql数据库中



执行结果



输入migrate命令，执行makemigrate生成的sql语句，表就建好了



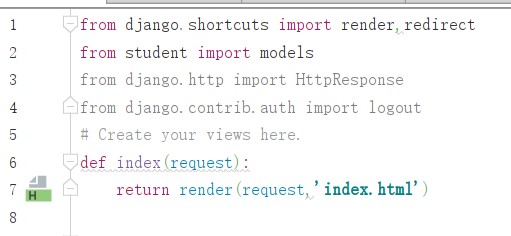
1. 创建模板



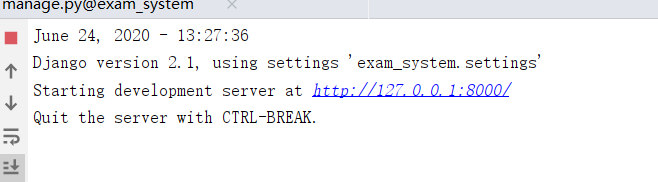


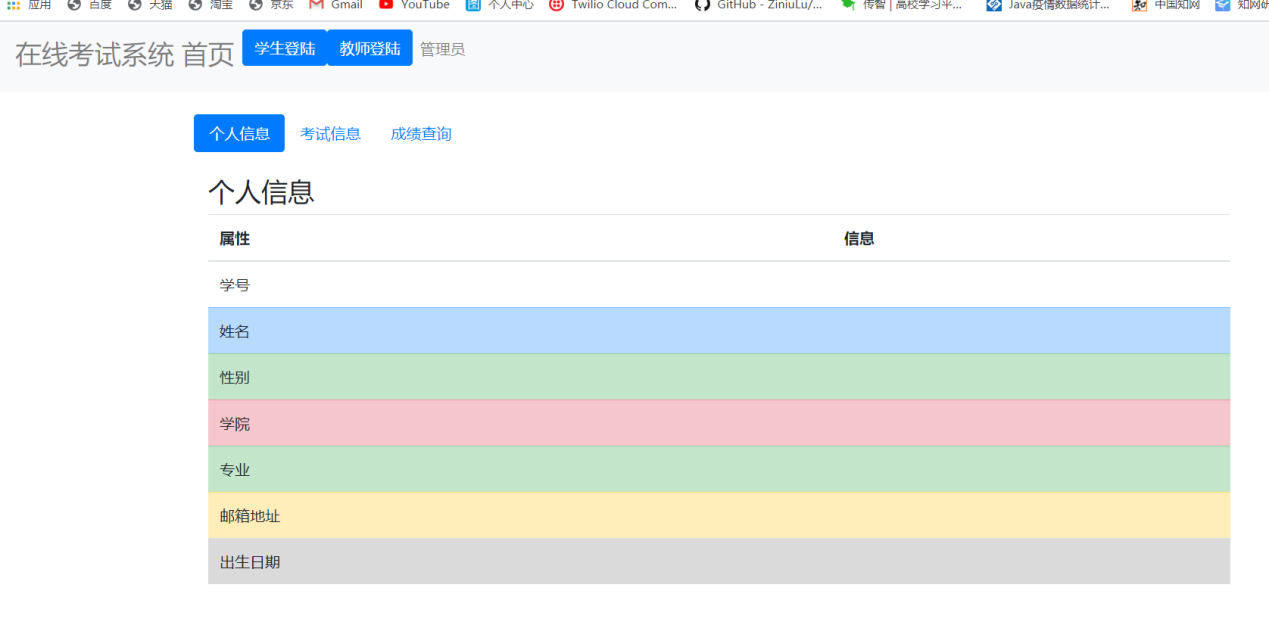


1. 创建视图函数



1. 启动服务器

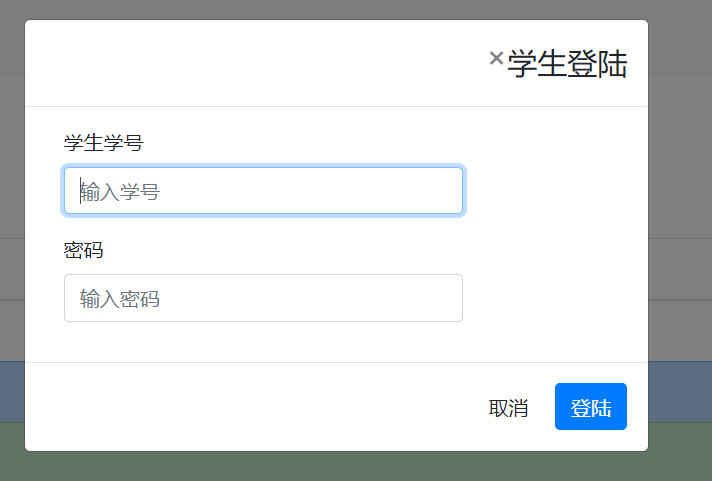




1. 各个模块的注册

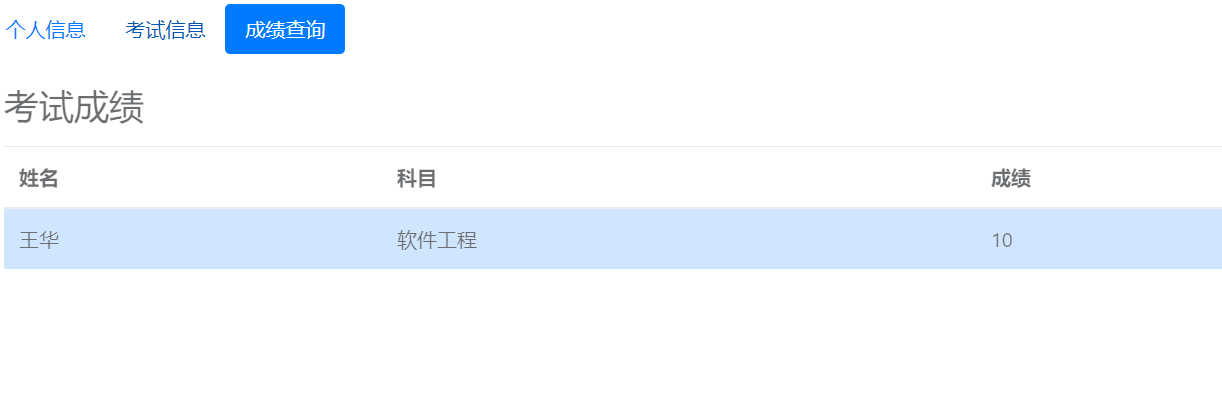
from django.contrib import admin  
from .models import Student,Teacher,Paper,Question,Grade  
# Register your models here.  
# 修改名称  
admin.site.site\_header=**'在线考试系统后台'**admin.site.site\_title=**'在线考试系统'**@admin.register(Student)  
class StudentAdmin(admin.ModelAdmin):  
 list\_display = (**'id'**,**'name'**,**'sex'**,**'dept'**,**'major'**,**'password'**,**'email'**,**'birth'**)# 要显示哪些信息  
 list\_display\_links = (**'id'**,**'name'**)#点击哪些信息可以进入编辑页面  
 search\_fields = [**'name'**,**'dept'**,**'major'**,**'birth'**] #指定要搜索的字段，将会出现一个搜索框让管理员搜索关键词  
 list\_filter =[**'name'**,**'dept'**,**'major'**,**'birth'**]#指定列表过滤器，右边将会出现一个快捷的过滤选项  
  
@admin.register(Teacher)  
class TeacherAdmin(admin.ModelAdmin):  
 list\_display = (**'id'**, **'name'**, **'sex'**, **'dept'**, **'password'**, **'email'**, **'birth'**)  
 list\_display\_links = (**'id'**, **'name'**)  
 search\_fields = [**'name'**, **'dept'**, **'birth'**]  
 list\_filter = [**'name'**,**'dept'**]  
  
@admin.register(Question)  
class QuestionAdmin(admin.ModelAdmin):  
 list\_display = (**'id'**,**'subject'**,**'title'**,**'optionA'**,**'optionB'**,**'optionC'**,**'optionD'**,**'answer'**,**'level'**,**'score'**)  
  
  
@admin.register(Paper)  
class PaperAdmin(admin.ModelAdmin):  
 list\_display = (**'tid'**,**'major'**,**'subject'**,**'examtime'**)  
 list\_display\_links = (**'major'**,**'subject'**,**'examtime'**)  
  
@admin.register(Grade)  
class GradeAdmin(admin.ModelAdmin):  
 list\_display = (**'sid'**,**'subject'**,**'grade'**,)  
 list\_display\_links = (**'sid'**,**'subject'**,**'grade'**,)  
 # fk\_fields = ['sid']

1. 实现学生登录







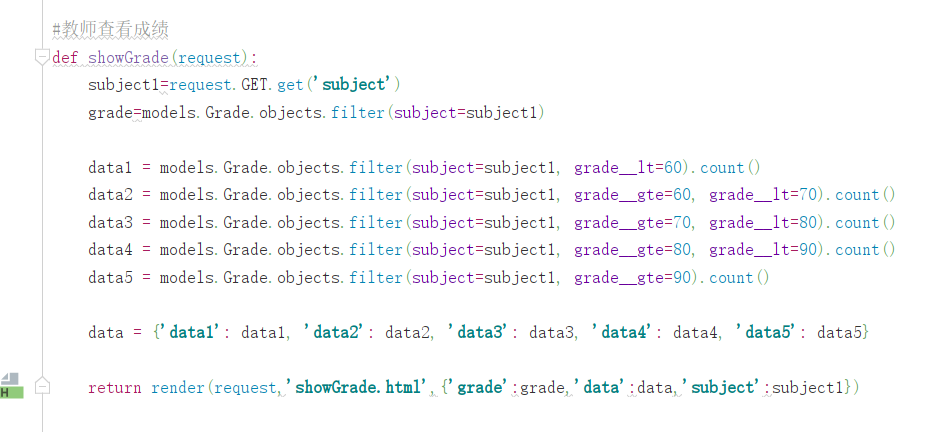
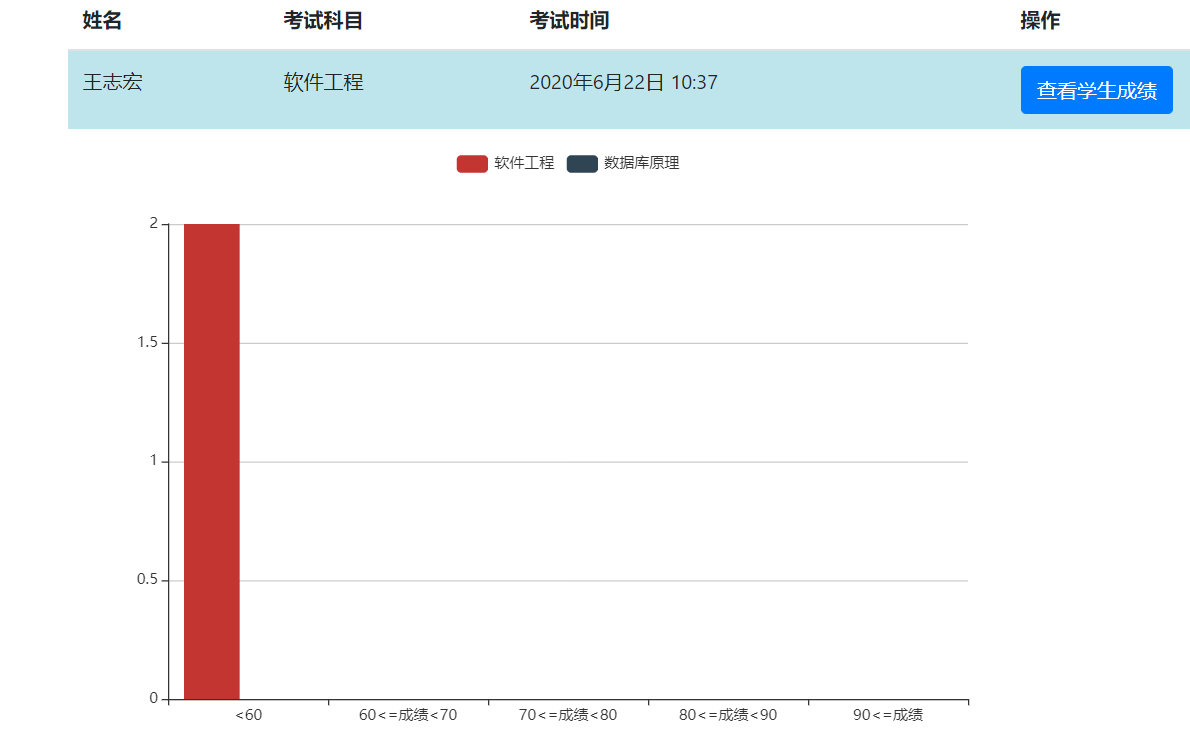
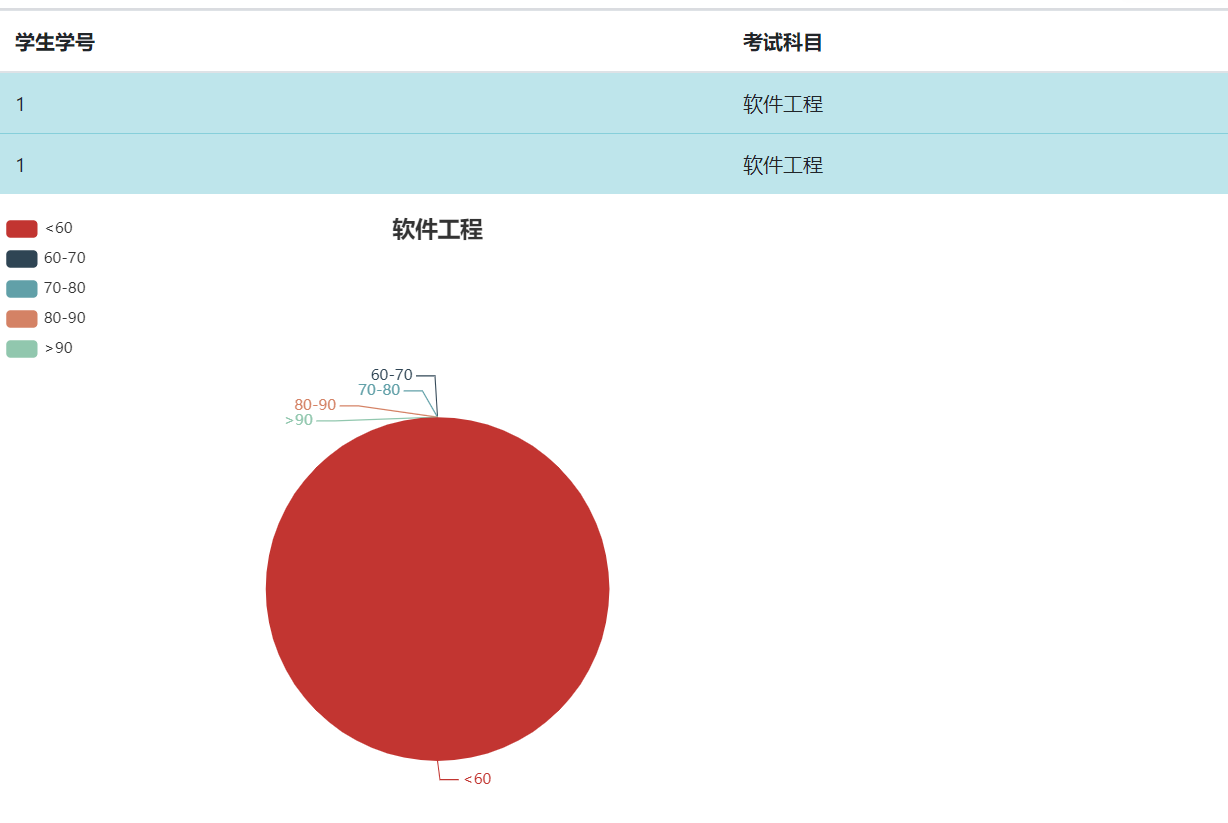
考试界面



1. 教师登录同上



教师查看成绩

# 5.测试过程

5.1链接测试

我们先是进行了前端页面的开发，接着对页面链接功能进行测试，通过点击不同的标题成功跳转到不同的页面。最后发现没异常。

5.2表单测试

经过后端的开发搭建完善后，我们进行了表单测试，发现数据能够很好的传递到服务器中，并插入到数据库中，实现了用户的登录功能。

5.3内容测试

每增加一个新功能时，我们都会进行内容测试，但是发现只有切换皮肤功能，在每次用户登录后，都会变回原型，于是我们通过采取本地存储的方式，来保存状态，最后发现还是不行，于是切换皮肤功能模块就成为了一个半成品。考生信息中的考试信息，最开始也是没能成功的显示在页面上，后来经过改变考试的科目类型，最后考试信息无法显示的情况就解决了。

5.4数据库测试

我们在对数据进行传输时，通过使用正则的方式，来对数据进行处理，让其能上传到数据库中，不会导致类型不匹配而产生错误。

# 6.验证需求

需求是在问题及其最终解决方案之间架设桥梁的一步。开发者只有和用户充分理解了需求之后才能开始设计系统，否则，对需求定义的任何改进，设计上都必须大量的返工。

6.1一致性需求

所有需求必须是一致的,任何一条需求不能和其他需求互相矛。

6.2 完整性需求

1. 所有输入的表单和参数在被软件处理之前，都需要根据允许的输入数据集进行比较验证，如身份的核实（需要图书管理员核实学生身份之后进行借书还书操作）等。
2. 所有发行的系统功能模块都应具有校验和、散列函数的功能，以便使用者能够验证该模块的准确性和完整性
3. 所有非人类的行动者（如系统和批处理程序）都需要被识别和监控，以防止它们对运行的系统数据进行操作，除非有明确的授权。

6.3现实性需求

1. 要能够抵御用户可能的误操作，保证软件的健壮性
2. 要对数据进行检验，保证数据有效性
3. 在数据被破坏时，具有数据恢复能力

# 总结

“互联网+教育”对师生来说也是一种很好的混合式学习方式和创新的教育模式。随着移动互联网应用的发展以及良好的用户体验，在线考试系统必然将在未来教育领域中发挥越来越多的作用，以达到与教育的深度融合。这种方式不仅成本低、效率高、便捷,同时辅助教师的教学,让教师和学生通在线考试的方式来进行考试，生成数据反馈给教师，教师对结果一目了然，提高了效率。

本论文所设计的在线考试系统,极具移动信息时代的特征,因此有着较好的前景和优势。但受限于目前的研究水平和开发能力,以及时间问题,本文设计的系统功能不够完善,还有很大的改善进步空间,值得继续深入研究探讨，可以在未来的工作中尝试去添加改善,成为教师教学上课所必须的综合型辅助工具。