数据库pj项目文档

17307130313 田嘉禾

17302010065 石睿欣

## 1.1项目概述

本项目旨在为学生选课，退课及教师查看课程，登记成绩，管理员管理课程提供方便的接口。项目实现的功能有：

学生：

1. 查看可选课程，已选课程，对课程进行筛选并进行选课
2. 退课
3. 进行选课申请并查看申请是否通过
4. 查看考试成绩及绩点

教师：

1. 查看教学的课程及课程花名册，对选课学生进行筛选
2. 处理选课申请
3. 登记成绩（手动或通过文件导入）

系统管理员（使用root为用户名和工号来登录系统）：

1. 批量导入学生，教师，课程，考试信息，并查看所有信息
2. 通过页面手动导入学生，教师，课程，考试信息，自动分配考试教室
3. 对以上信息进行更新及删除
4. 处理系统的状态，开启选课及登记成绩的权限

## 1.2 技术依赖

本项目基于MySQL与PHP结合实现后端，HTML+Jquery+Bootstrap实现前端完成。

## 1.3 功能特色

本项目在基本功能之外基于对用户友好的原则，对界面进行了一定的美化；

对于进行到一半因故退出的事务，利用JavaScript的localStorage进行数据本地持久化并在系统重启后能够恢复该事务。

对于进行到一半产生的数据冲突，将会根据冲突的数据类型不同给予用户提醒。

当因为数据冲突或系统故障插入失败时，基于原子性原则对事务进行了回滚。

MySQL系统内设置了管理员，老师，学生三种类型的用户并根据用户类型的不同设置不同类型的数据库连接，有效地防止了越权问题。

对于PHP访问MySQL的操作，使用用户输入的参数都利用了PHP MySQL预处理来防止SQL注入。

利用session持久化用户登录状态，并当用户访问越权页面时自动进行跳转。

对于学生界面的可选课程信息，教师界面的花名册信息及管理员界面的各类信息提供了筛选搜索的接口。

## 1.4 功能实现细节

具体可参照需求分析文档。

在此之上添加了系统的启动；登分；停止三种状态。

启动状态为学生可选课，退课，提交申请，教师可查看课程已选名单并处理申请；登分状态为教师登录成绩，学生查看成绩；停止状态下教师及学生用户不具有上述权限。

进入登分状态后，学生将无法继续选课，退课，发起选课申请。

## 2.1 数据库结构

数据库中各实体集表结构如下：

application：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| appli\_id | int(11) | NO | PRI |  | auto\_increment |
| appli\_status | varchar(20) |  |  |  |  |
| appli\_content | text |  |  |  |  |
| appli\_time | datetime |  |  | CURRENT\_TIMESTAMP |  |
| student\_id | varchar(4) | NO | MUL |  |  |
| sec\_id | int(11) |  | MUL |  |  |
| course\_id | varchar(4) | NO |  |  |  |
| semester  year | varchar(10)  int(11) |  |  |  |  |

classroom:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| classroom\_id | varchar(10) | NO | PRI |  |  |
| capacity | int(11) |  |  |  |  |

course:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| course\_id | varchar(4) | NO | PRI |  |  |
| course\_name  credit  class\_hours | varchar(100)  int(11)  int(11) | NO |  |  |  |

section:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| sec\_id | int(11) | NO | PRI |  |  |
| semester  year  start\_week  end\_week  number  selected\_num  course\_id  exam\_id | varchar(10)  int(11)  int(11)  int(11)  int(11)  int(11)  varchar(4)  int(11) | NO  NO  NO | PRI  PRI  PRI  MUL |  |  |

exam:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| exam\_id | int(11) | NO | PRI |  | auto\_increment |
| week | int(11) |  |  |  |  |

test:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| exam\_id | int(11) | NO | PRI |  |  |
| style | varchar(100) |  |  |  |  |

paper:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| exam\_id | int(11) | NO | PRI |  |  |
| demand | varchar(100) |  |  |  |  |

instructor:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| instructor\_id | varchar(4) | NO | PRI |  |  |
| instructor\_name  hire\_time  quit\_time | varchar(100)  varchar(10)  varchar(10) | NO |  |  |  |

student:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| student\_id | varchar(4) | NO | PRI |  |  |
| student\_name  total\_credit  gpa  enroll\_time  graduate\_time | varchar(100)  int(11)  double  varchar(10)  varchar(10) | NO |  |  |  |

time\_slot:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| time\_slot\_id | varchar(10) | NO | PRI |  |  |
| start\_time  end\_time  day\_of\_week | varchar(8)  varchar(8)  int(11) | NO  NO  NO |  |  |  |

注：另有class\_time\_place, exam\_time, exam\_time\_place, drops, takes, teaches, take\_exam关系集，关系集的字段为相关实体集的主键，takes关系集另有grade属性。

## 2.2 设计理念

总体参照教材上的University数据库设计。遵循了当两实体集的关系为一对多或多对一时将使用外键约束表示关系，而为多对多时创建关系集来表达关系的原则。另对其中几点加以说明：

1. exam\_time与exam\_time\_place分为两个关系集，方便导入考试时起初未安排考场，后来由系统自动安排考场的情况。
2. test与paper和exam间为继承关系，原因是与时间地点相关联的只有test类型的考试，因此单独列为实体集方便进行后续操作。
3. 对于主键以外的关键属性（如学生，教师姓名，时间段起始时间等）也使用了Not null来进行约束。
4. 在如class\_time\_place的关系集中对于time\_slot\_id与classroom\_id的组合添加了unique约束来防止同一时间段有多个课程使用教室，类似地还有exam\_time\_place关系集。