高级程序设计 课程

论文（报告、案例分析）

院 系 信息学院

专 业 信息管理与电子商务

班 级 152133003

学生姓名 王　朝 50%

学 号 1521330115

学生姓名 俄热空·哈孜太 25%

学 号 1521330094

学生姓名 郭亚楠 25%

学 号 1521330096

任课教师 张海军

2018年1月3日

课程论文评分表

|  |  |  |  |
| --- | --- | --- | --- |
|  | 各项  满分 | 评分标准 | 学生  得分 |
| 评分标准一 | 20 | 论文结构完整，包含论文题目、摘要、引文、正文、结论、参考文献 |  |
| 评分标准二 | 30 | 正文重点突出、结构布局严谨、逻辑性强 |  |
| 评分标准三 | 20 | 论文明确、证据充足、内容详实 |  |
| 评分标准四 | 10 | 文章中有独立见解与结论 |  |
| 评分标准五 | 10 | 语言流畅、叙述清楚 |  |
| 评分标准六 | 10 | 格式与参考文献规范、组数符合要求 |  |
| 总分 |  | | |
| 评阅人签字 |  | | |

学生考试管理系统设计

目录

[学生考试管理系统设计 3](#_Toc502771084)

[一、总体结构设计 4](#_Toc502771085)

[1老师模块划分 4](#_Toc502771086)

[2学生模块划分 4](#_Toc502771087)

[二、数据库逻辑结构设计--关系模式 5](#_Toc502771088)

[三、数据库逻辑结构设计--关系表 6](#_Toc502771089)

[1表单 6](#_Toc502771090)

[2教师信息表 6](#_Toc502771091)

[3学生信息表 6](#_Toc502771092)

[4试题信息表 7](#_Toc502771093)

[5、知识点表 7](#_Toc502771094)

[6参数表 7](#_Toc502771095)

[四、数据库表关系设计 8](#_Toc502771096)

[五、数据库行为设计 8](#_Toc502771097)

[六、逻辑模块设计 8](#_Toc502771098)

[1.1 用户登录 8](#_Toc502771099)

[1.2 用户退出 9](#_Toc502771100)

[2.1 教师添加知识点 9](#_Toc502771101)

[2.2 教师添加题目 9](#_Toc502771102)

[2.3 更改题数 10](#_Toc502771103)

[2.4 学生信息管理 10](#_Toc502771104)

[2.5 修改权限 11](#_Toc502771105)

[2.6 教师修改密码 11](#_Toc502771106)

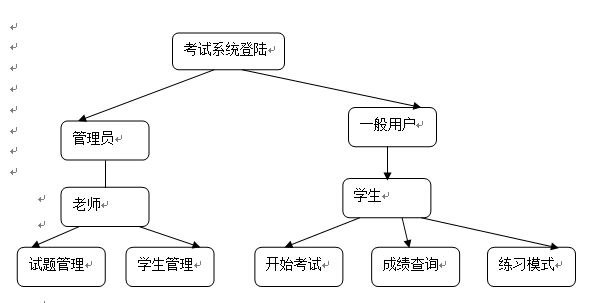
[3.1 学生进行题目练习 12](#_Toc502771107)

[3.2 学生进行考试 12](#_Toc502771108)

[3.3 学生成绩查询 13](#_Toc502771109)

[3.4 学生修改密码 13](#_Toc502771110)

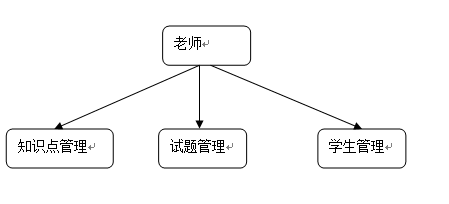
# 一、总体结构设计



总体结构设计

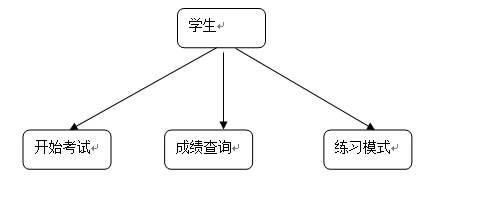
由总体结构图可知，本考试系统分为老师与学生两个模块，而老师模块又分为试题管理模块与学生管理模块。学生模块则分为考试模块，练习模块与成绩查询模块。

## 1老师模块划分

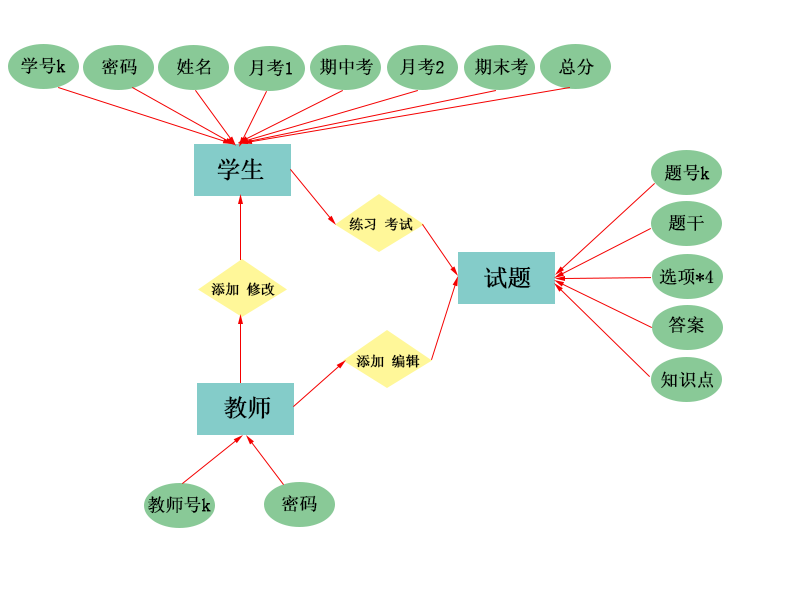


## 2学生模块划分

学生模块包括开始考试，成绩查询，与练习模式。开始考试模块包括试题的提交，下一题与退出键。成绩查询有退出键，练习模式包括试题的提交，下一题与退出键还有知识点的显示，如下图所示：



# 二、数据库逻辑结构设计--关系模式



E-R图

根据ER图的转换原则进行关系模式转换，得如下所示：

1、用户（用户名，身份,密码）

2、教师（教师编号（账号），密码，）

3、学生（学号，姓名，密码，成绩）

4、试题（题号，题干，选项，答案，知识点）

5、知识点（名称，章，节）

6、参数（权重1，权重2，权重3，权重4，考试题数）

7、教师和试题之间存在的关系（题号，题干，选项，答案，知识点）

8、学生和试题之间存在的关系（题号，答案，成绩）

# 三、数据库逻辑结构设计--关系表

## 1表单

|  |  |  |
| --- | --- | --- |
| 编号 | 名称 | 说明 |
| 表1-1 | Students | 学生信息表 |
| 表1-2 | Questions | 试题信息表 |
| 表1-3 | Teachers | 教师信息表 |
| 表1-4 | Knowledge | 知识点表 |
| 表1-5 | Canshu | 参数表 |

## 2教师信息表

教师信息表包含着教师信息和教师记录两种信息，教师在注册时需要填写自己的相关信息，包括用户名、账号、密码；另一个部分包含住用户记录，包括登录后进行的试题添加和修改，字段具体定义如下表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 教师表 | | | | |
| 字段名 | 类型 | 长度 | 允许为空 | 描述 |
| TID | varchar | 10 | 否 | 编号（主键） |
| TPassword | varchar | 10 | 否 | 密码 |

## 3学生信息表

学生信息表包含着学生基本信息和成绩，学生基本信息包括学生姓名，学号，密码；另一项则是学生的成绩。具体字段定义如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学生表 | | | | |
| 字段名 | 类型 | 长度 | 允许为空 | 描述 |
| SID | varchar | 10 | 否 | 学号（主键） |
| SPassword | varchar | 10 | 否 | 密码 |
| SName | nvarchar | 10 | 否 | 姓名 |
| SGrade1 | char | 3 | 是 | 月考1成绩 |
| SGrade2 | char | 3 | 是 | 期中考成绩 |
| SGrade3 | char | 3 | 是 | 月考2成绩 |
| SGrade4 | char | 3 | 是 | 期末考成绩 |
| SSumGrade | 函数 | 3 | 是 | 总成绩 |

## 4试题信息表

试题信息表包括题号，题干，选项，答案，知识点，分数。具体字段定义如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 试题表 | | | | |
| 字段名 | 类型 | 长度 | 允许为空 | 描述 |
| QID | varchar | 自动编号 | 否 | 题号（主键） |
| QQuestion | nvarchar | 200 | 否 | 题干 |
| QOptionA | nvarchar | 20 | 否 | 选项A |
| QOptionB | nvarchar | 20 | 否 | 选项B |
| QOptionC | nvarchar | 20 | 是 | 选项C |
| QOptionD | nvarchar | 20 | 是 | 选项D |
| QAnswer | 文本（选择） | A或B或C或D | 否 | 答案 |
| QKnowledgePiont | nvarchar | 200 | 是 | 知识点 |

## 5、知识点表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 知识点表 | | | | |
| 字段名 | 类型 | 长度 | 允许为空 | 描述 |
| KName | nvarchar | 10 | 否 | 名称 |
| KChapter | nvarchar | 2 | 否 | 章 |
| KSection | nvarchar | 2 | 否 | 节 |

## 6参数表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数表 | | | | |
| 字段名 | 类型 | 长度 | 允许为空 | 描述 |
| a | float | 10 | 否 | 考试权重1 |
| b | float | 10 | 否 | 考试权重2 |
| c | float | 10 | 否 | 考试权重3 |
| d | float | 10 | 否 | 考试权重4 |
| suma | int | - | 否 | 考试题数 |

# 四、数据库表关系设计

老师信息表保存的是对于老师的信息即教师登录账号和密码，学生则是包含学生信息包括学生登录账号密码以及学生各次考试的成绩，老师会将添加和修改的知识点和试题存放于知识点信息表和试题信息表，学生考试时从试题表中抽取题目，考试结束后将结果记录进学生表，参数表记录系统参数，包括考试权重及考试题数。

# 五、数据库行为设计

登录：通过选择不同角色（教师或学生），调用不同表，判断输入的账号密码是否正确，并作出反馈。

添加知识点和试题：教师通过系统读取、添加、修改、删除知识点和试题，并同步记录到知识点表和试题表。

管理学生：教师通过系统读取、添加、修改、删除学生信息进行学生管理，并同步记录到学生表中。

练习和考试：系统从试题表中读取试题，学生进行考试后将成绩记录进学生表。

修改密码：教师和学生调用并修改教师表和学生表中自己的信息，完成密码修改。

# 六、逻辑模块设计

### 1.1 用户登录

|  |  |
| --- | --- |
| 输入信息 | 用户名、密码、角色（选择） |
| 逻辑设计 | 1. 用户打开软件得到登录界面。 2. 用户填写用户名、密码并选择角色。 3. 点击“登录”，系统根据填写的信息，从“Teacher”或“Student”表中查找相应的用户名和密码，若能够匹配，用户登录成功，进入相应界面，若没有匹配项，则显示“用户名或密码错误！”。 4. 系统将用户名和密码赋予类username中的usrname和usrpassword临时保存。 |

### 1.2 用户退出

|  |  |
| --- | --- |
| 输入信息 | （无） |
| 逻辑设计 | 1. 学生和教师均可通过单击登陆后主界面的“退出登录”按钮回到“用户登录”界面。 |

### 2.1 教师添加知识点

|  |  |
| --- | --- |
| 输入信息 | 知识点名称、章、节 |
| 逻辑设计 | 1. 教师在教师端点击“考试管理”→“知识点管理”进入。 2. 点击表格空白处，在下面的空白处填写相应的知识点信息。 3. 点击“添加”按钮，系统检查Knowledge表中是否已有相同的知识点，如果有，报“知识点已存在！”错误，如没有，将用户填写的信息添加进Knowledge表中，并刷新界面。 4. 用户点击表格中某条已存在的知识点，其信息会显示在下方空白处，用户可以在此处修改信息，之后点击“修改”按钮，系统将根据知识点名称更新数据库数据，完成知识点信息的修改。 5. 用户点击表格中某条已存在的知识点，其信息会显示在下方空白处，点击“删除”按钮，系统会询问“是否删除名称为\*的知识点？”，单击“确认”，该知识点从数据库中删除。 6. 单击“退出”回到教师端主界面。 |

### 2.2 教师添加题目

|  |  |
| --- | --- |
| 输入信息 | 题干，选项A、选项B、选项C、选项D、答案、知识点 |
| 逻辑设计 | 1. 教师在教师端点击“考试管理”→“试题管理”进入。 2. 点击表格空白处，在下面的空白处填写相应的试题信息。 3. 点击“添加”按钮，系统检查Question表中是否已有相同的试题，如果有，报“该题目已存在！”错误，如没有，将用户填写的信息添加进Question表中，并刷新界面。 4. 用户点击表格中某条已存在的试题，其信息会显示在下方空白处，用户可以在此处修改信息，之后点击“修改”按钮，系统将根据题目编号更新数据库数据，完成试题信息的修改。 5. 用户点击表格中某条已存在的试题，其信息会显示在下方空白处，点击“删除”按钮，系统会询问“是否删除该题？”，单击“确认”，该题从数据库中删除。 6. 单击“更改题数”按钮，进入“更改题数”界面。 7. 单击“退出”回到教师端主界面。 |

### 2.3 更改题数

|  |  |
| --- | --- |
| 输入信息 | 题数 |
| 逻辑设计 | 1. 教师在教师端点击“考试管理”→“试题管理”→“更改题数”进入。 2. 在空格处输入想要作为考试中题目总数的数字。 3. 单击“确定”，系统从Question表中查询题目总数，并与输入数字比较，若输入数字大于总题数，报错：“超出总题数！”，否则将输入数字添加进canshu表的suma项，作为考试题数。 4. 单击“取消”回到“试题管理”界面。 |

### 2.4 学生信息管理

|  |  |
| --- | --- |
| 输入信息 | 学生学号、学生姓名、登录密码 |
| 逻辑设计 | 1. 教师在教师端点击“考试管理”→“试题管理”进入。 2. 点击表格空白处，在下面的空白处填写“学生学号”、“学生姓名“、”登录密码”三项。 3. 点击“添加”按钮，系统检查Student表中是否已有相同的学号，如果有，报“该学号已存在！”错误，如没有，将用户填写的信息添加进Student表中，并刷新界面。学生的四次考试成绩默认为空值。 4. 用户点击表格中某条已存在的学生条目，其信息会显示在下方空白处，用户可以在此处修改信息（学号无法修改），之后点击“修改”按钮，系统将根据学号更新数据库数据，完成学生信息的修改。 5. 用户点击表格中某条已存在的学生条目，其信息会显示在下方空白处，点击“删除”按钮，系统会询问“是否删除姓名为\*的学生信息？”，单击“确认”，该学生的学生信息从数据库中删除。 6. 单击“修改权重”按钮，进入“修改权重”界面。 7. 单击“退出”回到教师端主界面。 |

### 2.5 修改权限

|  |  |
| --- | --- |
| 输入信息 | 月考一成绩权重、期中考成绩权重、月考二成绩权重、期末考成绩权重 |
| 逻辑设计 | 1. 教师在教师端点击“考试管理”→“学生信息管理”→“修改权重”进入。 2. 在空格处输入想要作为考试分数所占总成绩权重的数字。 3. 单击“确定”，系统查看输入数字是否符合要求，若输入数字不符合要求，报错：“输入字符串格式不正确”；若四个数字加和不为“1”，报错“系数之和不为1”。若没有错误，则将四个输入的数字添加进canshu表的a、b、c、d项，作为四次考试的得分权重。 4. 单击“取消”回到“学生信息管理”界面。 |

### 2.6 教师修改密码

|  |  |
| --- | --- |
| 输入信息 | 原密码、新密码、确认密码 |
| 逻辑设计 | 1. 教师在教师端点击“修改密码”进入。 2. 在下方空格处输入“原密码”、“新密码”和“确认密码”。 3. 单击“确定”，系统查看输入密码是否符合要求，若“原密码”与username.usrpassword不同，报错“原密码错误”；若“新密码”与username.usrpassword相同，报错“新密码与原密码相同！”；若“新密码”和“确认密码”不同，报错“两次输入密码不同”。若没有错误，则将“新密码”中的记录添加到Teacher表的TPassword中。 4. 单击“取消”回到教师端界面。 |

### 3.1 学生进行题目练习

|  |  |
| --- | --- |
| 输入信息 | 选项（从A、B、C、D四个radiobutton中选择一个） |
| 逻辑设计 | 1. 学生在学生端点击“考试管理”→“练习模式”进入。此时系统从Question表中随机选择一条记录，将题干选项等显示在相应位置，答案保存在内存中。 2. 选择选项。 3. 单击“提交答案”，若没有选择，报错“请完成题目！”，否则将选中的radiobutton转换为相应的字母选项，与内存中的答案比较。如果二者相同，显示“答案正确”，二者不同，显示“答案错误，争取答案为\*”。 4. 单击“下一题”，系统重新完成1.的内容，出现一道新题 5. 单击“退出练习”回到学生端界面。 |

### 3.2 学生进行考试

|  |  |
| --- | --- |
| 输入信息 | 考试名称、选项 |
| 逻辑设计 | 1. 学生在学生端点击“考试管理”→“正式考试”进入。此时系统会让学生选择考试的名称，学生进行选择后，系统从Student表中查看学生信息，若此学生此次考试成绩为空，允许考试，否则显示错误：“你已参加过这次考试”。 2. 允许考试的学生进入考试界面，系统从Question表中随机选取canshu.suma道试题存入内存。 3. 学生选择选项，当学生单击“上一题”或“下一题”时，学生的选项转化为字母暂时存储在一个有canshu.suma个元素的数组中。 4. 学生考试完毕，点击“提交试卷”，系统通过该数组中是否有空值以及空值的位置告诉学生是否有题目未做完以及未做完题目的题号。若数组元素都不为空，系统通过循环语句，将数组元素与临时表中Qanswer列中的答案进行对比，每匹配一个，成绩myscore（原值为0的int）+1，最后学生的成绩为myscore/canshu.suma转换为百分制。 5. 向学生显示“你的成绩为\*”并自动退回到学生端主界面。 |

### 3.3 学生成绩查询

|  |  |
| --- | --- |
| 输入信息 | （无） |
| 逻辑设计 | 1. 学生在学生端点击“账号管理”→“成绩查询”进入。 2. 系统从Student表中读取该学生的SGradeA-D，并显示在相应位置，总成绩由成绩和类quanzhong中的值a、b、c、d共同计算得到。 3. 单击“确定”返回学生端主界面。 |

### 3.4 学生修改密码

|  |  |
| --- | --- |
| 输入信息 | 原密码、新密码、确认密码 |
| 逻辑设计 | 1. 学生在学生端点击“账号管理”→“修改密码”进入。 2. 在下方空格处输入“原密码”、“新密码”和“确认密码”。 3. 单击“确定”，系统查看输入密码是否符合要求，若“原密码”与username.usrpassword不同，报错“原密码错误”；若“新密码”与username.usrpassword相同，报错“新密码与原密码相同！”；若“新密码”和“确认密码”不同，报错“两次输入密码不同”。若没有错误，则将“新密码”中的记录添加到Student表的SPassword中。 4. 单击“取消”回到学生端界面。 |