华中科技大学网上选课系统

数据库设计说明书

**V1.0**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 人员 | 时间 | 备注 |
| 编写 | 贾智勇、徐可辰 | 2023.4.21 |  |
| 审核 | 贾智勇、徐可辰 | 2023.4.21 |  |
|  |  |  |  |
|  |  |  |  |

1引言 3

1.1编写目的 3

1.2背景 3

1.3定义 3

1.4参考资料 3

2外部设计 4

2.1标识符和状态 4

2.2使用它的程序 4

2.3约定 4

2.4专门指导 4

2.5支持软件 5

3结构设计 5

3.1概念结构设计 5

3.1.1系统分E-R图 5

3.1.2系统总体E-R图 8

3.2逻辑结构设计 9

3.2.1基本表 9

3.3物理结构设计 11

3.3.1步骤 12

3.3.2存取方法 12

4运用设计 12

4.1数据字典设计 12

4.2安全保密设计 13

1引言

1.1编写目的

数据库的设计及是在功能性需求和非功能性需求明确之后，并且在详细设计文档里进行了各个模块的设计，以及如何实现，与上下层的接口和各个模块之间如何传递信息等的基础上参考概要设计文档进行的。

数据库设计说明书主要说明数据库的三级结构，包括概念结构设计，逻辑结构设计以及物理结构设计等。同时也说明了数据字典的设计。该设计文档主要是在编码开发阶段，指导开发人员如何进行数据库的搭建与实现，同时在测试阶段，该文档也有重要的参考价值。

该文档的预期读者包括：开发人员，测试人员。

1.2背景

任务提出者：贾智勇、徐可辰

用户：华中科技大学在校师生及管理人员

待开发系统名称：学生网上选课系统

待开发数据库名称：选课系统数据库

计算站：华中科技大学网络与计算中心

1.3定义

SQL（Structured Query Language）：结构化查询语言，一种特殊目的的编程语言，是一种数据库查询和程序设计语言，用于存取数据以及查询，更新和管理关系数据库系统，同时也是数据库脚本文件的扩展名。

MySQL（My Structured Query Language）：关系数据库管理系统，在web应用方面，MySQL是最好的RDBMS应用软件，具有体积小，速度快，成本低，开放源码并为多种编程语言提供了API等特点。

foreign key：外键，用于与另一张表相关联，是能确定另一张表记录的字段，用于保持数据的一致性。

Key：主键，唯一标识一条记录，不可有重复和空的数据项。

Index：索引，用以提高查询排序的速度。

IDEA：全称IntelliJ IDEA，是java语言开发的集成环境。

1.4参考资料

[1] 软件工程. （英）萨默维尔著，程成，陈霞译. 机械工业出版社 原书第九版

[2] 软件文档写作. 陈长清著. 清华大学出版社 第一版

[3] 构造之法 邹欣著 人民邮电出版社 第三版

[4] 华中科技大学现用网上选课系统

[5] 数据库设计说明书（GB8567—88）

2外部设计

2.1标识符和状态

数据库软件：MySQL

系统要求所创建的数据库的名称：course

2.2使用它的程序

本系统是B/S分成结构，将系统的核心功能集中到了服务器上，客户通过浏览器便可以轻松的实现对于应用程序的访问。

因此使用本系统的应用程序有：学生网上选课系统（1.0）

网上测试软件系统（1.o）

2.3约定

在本系统的数据库设计中，我们采用了一致命名法对数据库中表明和字段名进行命名。我们使用了MySQL数据库对数据进行存储和管理。下面对其进行描述：

1：学生网上选课系统的数据库设计过程中，主要涉及到的实体有：学院选课信息表，课程信息表，学生课程信息表，专业信息表，专业课程信息表，学生信息表，教师信息表，教师课程信息表，学院选课时间信息表等。其中我们用全英文为各个信息表命名，在单项信息表中，我们加info后缀来表示信息，如teacherinfo，studentinfo，majorinfo，courseinfo等，在两个信息相对应的信息表中，我们用\_to\_结构来表示他们之间的对应关系，如academytocourse，majortocourse，stutocourse，teachertocourse，timetoacademy等。整体采用这两种结构的命名方式，清晰明了，简单易懂。

2：对数据库中各个表的字段进行命名时，我们都使用全英文的方式，即英文单词或缩略单词对应中文词组，并且在特殊的字段，比如ID号，会加上\_ID作为后缀，如teacherID，courseID，academyID，stuID等，使用这种方式表达，使字段命名有意义，便于后续编码实现。

2.4专门指导

系统使用DBConnection实现与数据库的接口，该接口生成CLASS文件实现对数据库的建立和控制，是应用程序与数据库的桥梁。最终通过对CLASS文件操作来实现对数据库的管理工作。

2.5支持软件

Navicat for SQL：强大的数据库管理及开发工具，可快速建立数据库。

MySQL：关系数据库管理系统。

Idea：全称IntelliJ IDEA，是java语言开发的集成环境。

3结构设计

3.1概念结构设计

我们在需求分析阶段产生的需求说明的基础上，按照特定的方式把任务抽象为一个不依赖于任何具体机器的数据模型，也就是概念模型。其描述系统中的各个实体以及相关实体之间的关系，是系统特性和静态描述。

在我们得数据库系统中一共有六个实体：学生，教师，管理员，学院，专业，课程。

3.1.1系统分E-R图

1. 学生的属性：学号，姓名，性别，出生日期，密码，学院，专业，年级，已修学分

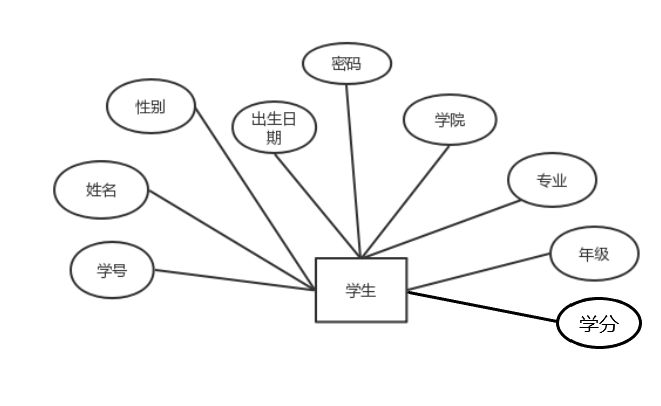


图3-1学生E-R图

1. 教师的属性：工号，姓名，性别，出生日期，密码，职称，院系

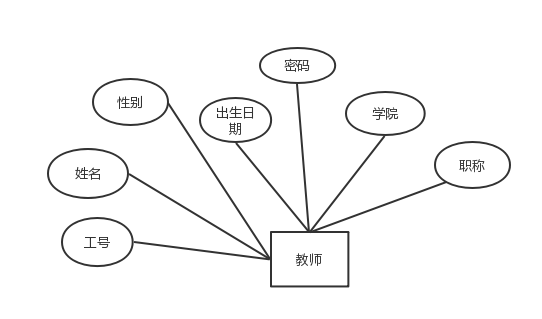


图3-2教师E-R图

1. 管理员的属性：工号，姓名，性别，出生日期，密码，权限

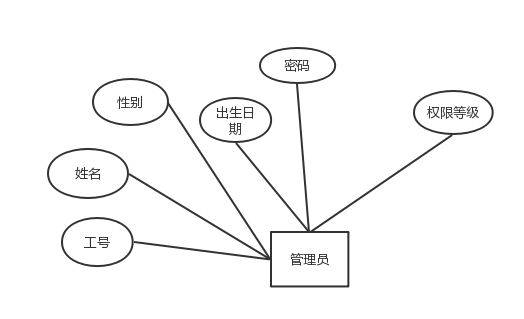


图3-3管理员E-R图

1. 专业的属性：专业号，专业名，联系方式，专业简介，专业负责人，专业等级，学分要求

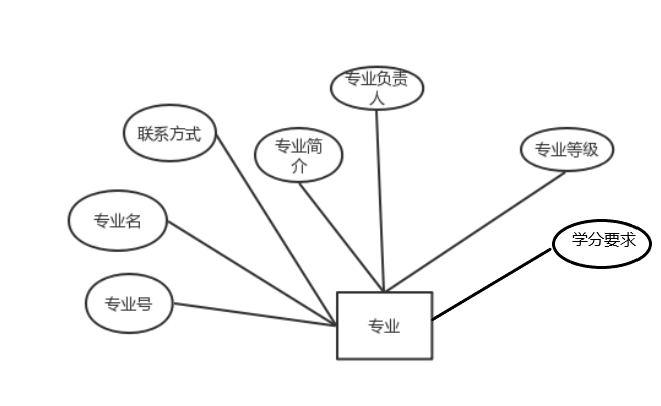


图3-4专业E-R图

1. 院系的属性：院系号，院系名，联系方式，院长，简介，学院等级

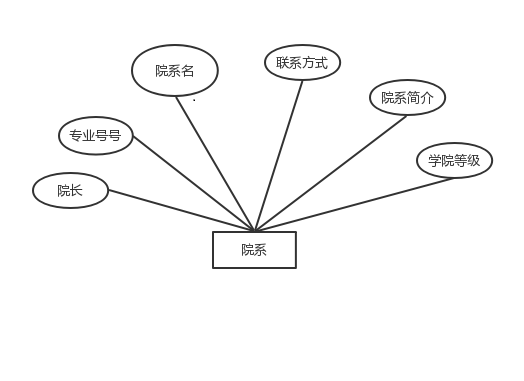


图3-5学生E-R图

1. 课程的属性：课程号，课程名，学时，学分，课程简介，开课学院

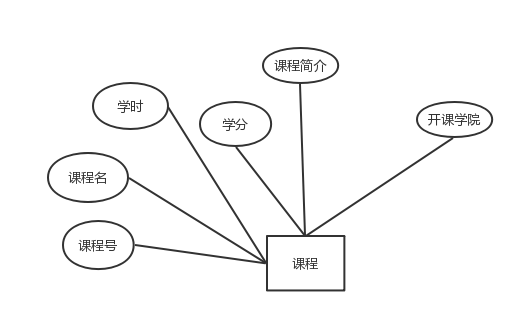


图3-6课程E-R图

1. 控制属性：选课时间控制，系统开闭控制，成绩录入控制

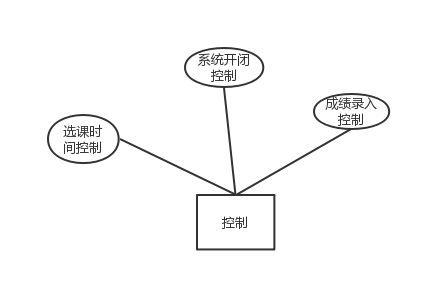


图3-7控制E-R图

3.1.2系统总体E-R图

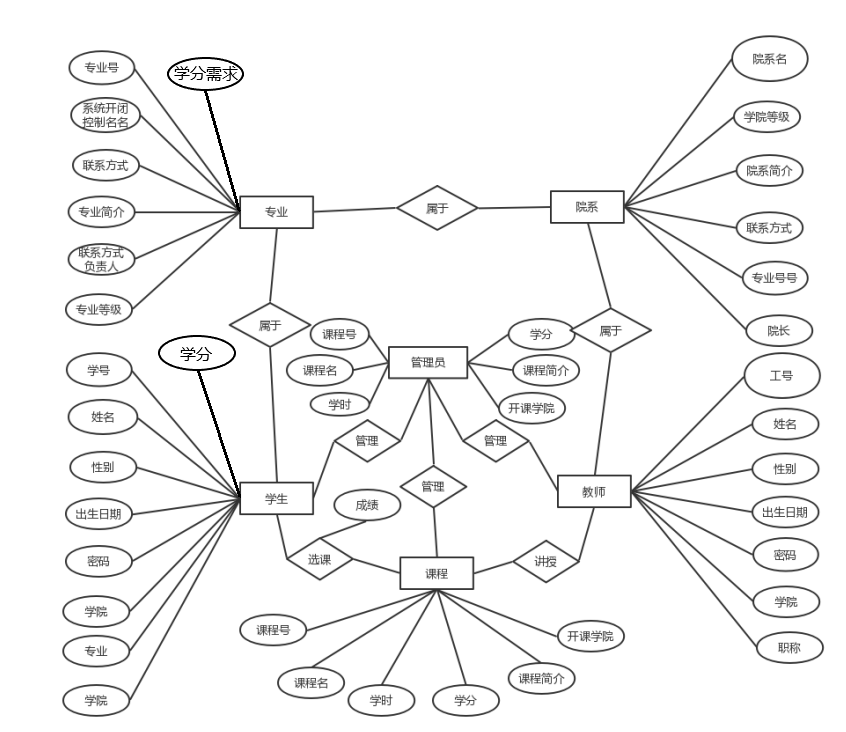


图3-8系统总体E-R图

3.2逻辑结构设计

3.2.1基本表

说明把上述原始数据进行分解、合并后重新组织起来的数据库全局逻辑结构，包括所确定的关键字和属性、重新确定的记录结构和文卷结构、所建立的各个文卷之间的相互关系，形成本数据库的数据库管理员视图。

逻辑结构对应概念模式，概念模式又称模式或者逻辑模式，对应于概念。他包括了本系统综合的所有用户的所有数据，按照统一的观点构造的全局逻辑结构，是所有用户的公共数据视图。本系统中所包括的表有学生信息表，教师信息表，课程信息表，学生选课记录表，学生已修课程记录表，学院课程对应表等等。他包括了本系统中所有用户所需要的数据以及对数据的描述，数据之间关系的描述。

表名：**academytocourse**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| academyID | 学院号 | char | NOT NULL |  |
| courseID | 课程编号 | char | NOT NULL |  |

表名：**classinfo**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| courseID | 课程号 | varchar | NOT NULL | PRIMARY KEY |
| courseName | 课程名 | varchar | DEFAULT NULL |  |
| teacherName | 任课老师 | varchar | DEFAULT NULL |  |
| quarter | 上课季度 | varchar | DEFAULT NULL |  |
| week | 上课周次 | varchar | NOT NULL | PRIMARY KEY |
| time | 上课时间 | varchar | NOT NULL | PRIMARY KEY |
| place | 上课地点 | varchar | NOT NULL | PRIMARY KEY |
| studentsNum | 学生数目 | bigint | DEFAULT NULL |  |

表名：**completeedcourse**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| stuID | 学生号 | char | NOT NULL |  |
| courseID | 课程号 | char | NOT NULL |  |
| courseName | 课程名 | varchar | NOT NULL |  |
| score | 分数 | float | DEFAULT NULL |  |
| pass | 是否通过 | char | DEFAULT NULL |  |
| teacherName | 教师姓名 | varchar | DEFAULT NULL |  |
| credit | 学分 | float | DEFAULT NULL |  |
| quarter | 季度 | varchar | DEFAULT NULL |  |
| hours | 小时数 | varchar | DEFAULT NULL |  |
| classification | 分类 | varchar | DEFAULT NULL |  |

表名：**courseinfo**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| courseID | 课程号 | varchar | NOT NULL | PRIMARY KEY |
| courseName | 课程名 | varchar | DEFAULT NULL |  |
| credit | 学分 | bigint | DEFAULT NULL |  |
| classification | 分类 | varchar | DEFAULT NULL |  |
| capacity | 教室容量 | bigint | DEFAULT NULL |  |
| hours | 小时 | varchar | DEFAULT NULL |  |

表名：**majorinfo**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| majorID | 专业编号 | varchar | NOT NULL | PRIMARY KEY |
| majorName | 专业名称 | varchar | DEFAULT NULL |  |
| academyID | 学院编号 | varchar | DEFAULT NULL |  |
| academyName | 学院名称 | varchar | DEFAULT NULL |  |
| credit | 学分需求 | bigint | DEFAULT NULL |  |

表名：**majortocourse**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| majorID |  | varchar | DEFAULT NULL |  |
| courseID |  | varchar | DEFAULT NULL |  |
| gradeNum |  | varchar | DEFAULT NULL |  |

表名：**studentinfo**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| stuID | 学生号 | varchar | NOT NULL | PRIMARY KEY |
| stuName | 学生姓名 | varchar | DEFAULT NULL |  |
| password | 密码 | varchar | DEFAULT NULL |  |
| majorID | 专业号 | varchar | DEFAULT NULL |  |
| gradeNum | 年级号 | varchar | DEFAULT NULL |  |
| classNum | 学号 | bigint | DEFAULT NULL |  |
| credit | 已修学分 | bigint | DEFAULT NULL |  |

表名：**stutocourse**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| stuID | 学生号 | char | NOT NULL |  |
| courseID | 课程号 | char | NOT NULL |  |
| classification | 分类 | varchar | NOT NULL |  |

表名：**teacherinfo**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| teacherID | 教师号 | varchar | NOT NULL | PRIMARY KEY |
| teacherName | 教师姓名 | varchar | DEFAULT NULL |  |
| password | 密码 | varchar | DEFAULT NULL |  |
| level | 等级 | varchar | DEFAULT NULL |  |

表名：**teachertocourse**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| teacherID | 教师号 | varchar | DEFAULT NULL |  |
| courseID | 课程号 | varchar | DEFAULT NULL |  |

表名：**timetoacademy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段描述 | 类型 | 是否允许为空 | 约束 |
| academyID | 学院号 | char | NOT NULL |  |
| classification | 分类 | varchar | NOT NULL |  |
| starttimemonth` | 开始月份 | int | DEFAULT NULL |  |
| starttimeday | 开始日期 | int | DEFAULT NULL |  |
| endtimemonth | 结束月份 | int | DEFAULT NULL |  |
| endtimeday | 结束日期 | int | DEFAULT NULL |  |

3.3物理结构设计

系统数据的物理结构设计主要包括数据库的在物理设备上的存储结构与存储方法等。为一个给定的逻辑数据模型选取一个最适合应用环境的物理结构的过程，就是数据库的物理设计。 对于本系统来说，由于本系统采取的是关系型数据库，而且数据库管理系统是MySQL，所以本系统采用索引存储方法作为关系模式的存取方法。

从数据库管理系统的角度来看，视图是对于数据类型的一种描述，当创建了一个视图的时候，便通过封装一个SELECT语句来定义元数据，当在另一个查询的 FROM 子句中引用视图时，将从系统目录检索该元数据，并替代该视图的引用扩展元数据。视图扩展之后，My SQL的查询优化器会为执行查询编译一个执行计划。本系统的查询条件所涉及到的属性如下：

1. 通过学号查询学生信息
2. 登录通过学号密码查找学生信息
3. 通过专业年级查找专业课
4. 通过学号，课程号，课程类型选修课程
5. 通过学号和课程类型查看选课结果
6. 通过学生学号和课程号退选课程
7. 通过学生学号查看已修课程
8. 通过学院编号查看可选的体育课
9. 通过专业号和课程类型查看选课时间
10. 通过类别查找公选课
11. 通过学号查询并计算已修学分和剩余学分

|  |  |
| --- | --- |
| 表名 | academytocourse |
| 索引字段 | 无 |
| 存取方式 | 默认 |

|  |  |
| --- | --- |
| 表名 | classinfo |
| 索引字段 | teacherName |
| 存取方式 | B+树索引 |

|  |  |
| --- | --- |
| 表名 | completeedcourse |
| 索引字段 | stuId |
| 存取方式 | B+树索引 |

|  |  |
| --- | --- |
| 表名 | courseinfo |
| 索引字段 | credit |
| 存取方式 | 聚簇索引 |

|  |  |
| --- | --- |
| 表名 | majorinfo |
| 索引字段 | 无 |
| 存取方式 | 默认 |

|  |  |
| --- | --- |
| 表名 | majortocourse |
| 索引字段 | majorID,courseID |
| 存取方式 | hash索引 |

|  |  |
| --- | --- |
| 表名 | studentinfo |
| 索引字段 | 无 |
| 存取方式 | 默认 |

|  |  |
| --- | --- |
| 表名 | stutocourse |
| 索引字段 | 无 |
| 存取方式 | 默认 |

|  |  |
| --- | --- |
| 表名 | teacherInfo |
| 索引字段 | 无 |
| 存取方式 | 默认 |

|  |  |
| --- | --- |
| 表名 | teachertocourse |
| 索引字段 | teacherID |
| 存取方式 | 聚簇索引 |

|  |  |
| --- | --- |
| 表名 | timetoacademy |
| 索引字段 | 无 |
| 存取方式 | 默认 |

4运用设计

4.1数据字典设计

数据字典是进行详细的数据收集和数据分析获得的主要成果。它是关于数据库中数

据的描述，即元数据，而不是数据本身。数据字典是在需求分析阶段建立，在数据设计

过程中不断修改、充实、完善的。它在数据库设计中占有很重要的地位。

数据字典通常包括数据项、数据结构、数据流、数据存储和数据处理过程几部分。

其中数据项是数据的最小组成单位，若干个数据项可以组成一个数据结构。数据字典通

过对数据项和数据结构的定义来描述数据流、数据存储的逻辑内容。

1. 数据结构名：学生数据结构

含义说明：学生选课系统的主体数据结构，定义了一个学生的有关信息

组成：学号，姓名，院系，专业，年级

1. 数据结构名：教师数据结构

含义说明：教师功能模块的主体数据结构，定义了一个教师的有关信息

组成：教师号，教师姓名，所在院系，职称

1. 数据结构名：课程数据结构

含义说明：学生可选课程的说明

组成：课程号，课程名，开课学院，学时，学分，课程简介

4.2安全保密设计

在学生网上选课系统中，由于用户的身份不同，对数据库的访问权限也会不同。系统定义学生之间隐私信息不可相互访问，教师之间以及学生教师之间隐私信息也不可相互访问。管理员几乎能够对所有用户一定表进行操作，包括增删改查等。根据实际情况，学生一旦选课成功，学生本人及管理员均不可对其选课信息再次作出修改。同样，学生与教师对数据的访问权限有限。这样在一定程度上保持数据库的安全性。另一方面，在数据库的各个表的结构设计中国，各表之间根据实体完整性，参照完整性以及用户自定义完整性等，设置了不同等级的约束。同时，系统会自动备份重要或者敏感数据，也支持管理员手动备份。

1：开发人员对系统的数据库数据，数据和浏览器之间的数据传输，用户的请求由过滤检测和防护，防止非法操作和数据泄露、

2：使用密码、数字图片验证码提供身份验证，以保证访问的用户不是机器，且确实是它所声称的身份。使用密码、数字图片验证码提供身份验证，以保证访问的用户不是机器，且确实是它所声称的身份。并且，系统对于通过身份验证的用户进行访问控制，赋予不同身份的用户不同的访问和操作权限。

3.对于关键数据，如学生成绩，自动进行修改记录，确保能够追溯关于修改行为的信息。