**摘要**

这次的课设我们做的是宾馆管理系统，采用的是PHP与MySQL的组合，在这个程序里面我们用到了很多PHP的函数以及HTML5的代码，HTML5主要是作为前端显示和美化的，主要的功能还是后端的PHP实现的，链接MySQL的方法用的是PHP自带的函数实现的，这个系统主要实现了对客户添加删除，以及结算等，还有对客房的添加和删除和对职工的管理等。还有以进度条的方式显示了宾馆的住房率，以及用不同的颜色标记了状态不同的房间，这样可以使人员能一眼就了解宾馆的住宿情况。该系统可以满足小型的宾馆管理，帮助宾馆管理人员更好更方便的管理宾馆。

**关键词：**宾馆住宿系统；PHP；数据库

**目录**

[1系统简介 1](#_Toc503437319)

[1.1系统开发的背景和意义 1](#_Toc503437320)

[1.2系统所要实现功能 1](#_Toc503437321)

[1.3系统开发工具 1](#_Toc503437322)

[1.4系统模块分析 2](#_Toc503437323)

[2需求分析 3](#_Toc503437324)

[2.1 需求调查 3](#_Toc503437325)

[2.2 软件功能需求 3](#_Toc503437326)

[2.3 数据处理需求 3](#_Toc503437327)

[2.4 数据需求 6](#_Toc503437328)

[2.4.1 数据项分析汇总 6](#_Toc503437329)

[2.4.2 数据整合与分类 6](#_Toc503437330)

[2.4.3 数据字典 7](#_Toc503437331)

[3数据库概念结构设计 10](#_Toc503437332)

[4 数据库逻辑结构设计 13](#_Toc503437333)

[4.1实体所对应的关系模式 13](#_Toc503437334)

[4.2关系模式优化处理 13](#_Toc503437335)

[4.3 数据表设计 13](#_Toc503437336)

[5 数据库物理结构设计 15](#_Toc503437337)

[5.1 系统采用的数据库简介 15](#_Toc503437338)

[5.2 数据库的建立和结构设计 15](#_Toc503437339)

[5.3 数据库的完整性设计 17](#_Toc503437340)

[5.4存储方法设计 19](#_Toc503437341)

[5.5存取结构设计 19](#_Toc503437342)

[5.6查询优化 20](#_Toc503437343)

[6 编码实现 21](#_Toc503437344)

[6.1 系统详细功能 21](#_Toc503437345)

[6.2 系统开发语言简介 22](#_Toc503437346)

[6.3 系统的实现 23](#_Toc503437347)

[6.4 数据库连接代码及数据操作代码 38](#_Toc503437348)

[7 测试分析报告 40](#_Toc503437349)

[总结 41](#_Toc503437350)

[参考文献 42](#_Toc503437351)

[致谢 43](#_Toc503437352)

# 1系统简介

## 1.1系统开发的背景和意义

随着我国经济的迅速发展，人们的生活水平有了显著的提高，假日经济和旅游经济已成为人们消费的热点。各地也把旅游业当作本地经济发展的重要经济支柱之一，从而促进了宾馆宾馆业的快速发展。同时，随着宾馆宾馆越来越多，人们的要求越来越高，宾馆宾馆业的竞争也愈来愈激烈。

如何在激烈的竞争中生从发展，是每一个宾馆宾馆必须面临的问题。对宾馆的经营状况起决定作用的是宾馆的服务管理水平。如何利用先进的管理手段、提高宾馆的管理水平，以成为宾馆宾馆业发展的当务之急。面对信息时代的机遇和挑战，利用科技手段提高企业管理水平是企业管理无疑的一条行之有效的途径。现在的宾馆行业已经发生拉巨大变化，已不仅是提供给人们一个吃住行的简单场所，所以，将计算机管理引入宾馆的管理体系中,无疑对宾馆的服务水平更上一层楼提供了一种有利的支持。

虽然计算机管理并不是宾馆管理走向成功的关键元素，但它可以最大限度地发挥准确、快捷、高效等作用，对宾馆的服务管理提供强有力的支持。因此，采用全新的计算机网络和宾馆管理系统，已成为提高宾馆的管理效率、改善服务的水准的重要手段之一。在某种意义上，宾馆客房管理方面的信息话已成为现代化宾馆的重要标志。

## 1.2系统所要实现功能

该系统应充分利用信息技术提高宾馆客房的管理水平、服务水平。宾馆客房管理系统要实现客房住宿的登记、宿费提醒、调房登记、退房结帐、查询统计等功能。该系统能详细的记录各种信息，客人从入住开始登记入住时间，房间号，房间类型，电话号码等，到客人离开会自动总结费用，显示截至日期，以及入住时的信息。除此之外还能实现对内部人员的管理，添加，查询，修改，删除内部职工，给宾馆的管理带来了方便。

## 1.3系统开发工具

开发数据库管理信息系统需要选择两种工具，即前台开发语言和后台数据库。本系统运用了PHPstorm作为开发工具，MYSQL作为后台数据库。

## 1.4系统模块分析

（1）系统管理模块

该模块实现操作员的管理,如添加删除和修改操作,并设置操作员的操作权限对指定的用户进行密码修改.

（2）客房管理模块状

该模块实现对客房的基本信息的设置,如客房的房间号，房间类型，房间价格等，还可以查询相应房间客人的信息。

（3）订房管理模块状

该模块实现用户订房信息管理,提供对订房客户资料的登记,查询和退订操作。其中订房信息包括客房的类型和订房的时间等。

（4）收银管理模块状

该模块实现用户的确订房信息管理,提供对订房客户资料的登记,查询，并提供当日报表的打印。

# 2需求分析

## 2.1 需求调查

该宾馆住宿系统需要满足以下要求：

客房管理（状态、利用率、等级、管理员）；

客房登记、入住和退房结算的流程管理；

房间基本信息表；

职工基本信息表；

客人信息表；

入住信息表。

## 2.2 软件功能需求

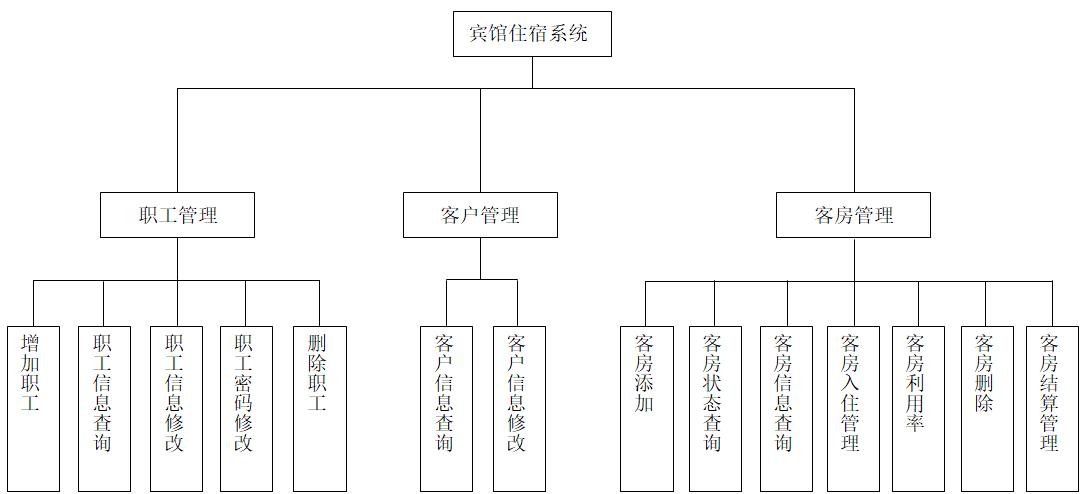
****

图2.1软件功能图

## 2.3 数据处理需求

宾馆在运营中需要对宾馆资源、顾客信息、结算信息进行管理，如果有顾客来往，经过对客房状态的查询，查看是否有房间能及时为客户办理登记手续，如果有顾客来退房，能及时为顾客登记并办理退房手续，并且修改客房信息，因此，利用开发完成后的宾馆管理系统能够十分便利的对住房系统进行管理有利于提高效率。

宾馆住房管理系统开发的任务是实现各种住房信息的系统规范化、自动化主要完成的功能有以下几个方面。

（1）住宿管理

这个模块包括三个部分，即住宿登记、调房登记、退宿结帐。该模块要实现客人的住宿登记，详细记录客人的住宿信息，如房间号，房间类型，电话号码，房间价格等，其后还可以进行房间调节，会修改住宿信息。最后在客人退房的时候自动结算，会显示客人的入住信息，住宿天数，及费用。

（2）客房管理

这个模块包括三个部分，即客房添加、客房查询、房态删除。宾馆客房在不同的时间有不同的状态，当客人来入住时，找到空闲房，有人入住，房间号不能再次利用，客人离开时可以删除已用的房间。

（3）查询修改管理

这个模块可以对客人查询修改，也可以对职工查询修改，当客人需要调房或者信息错误时可以查到并修改信息，职工的也一样，有更高权限的经理可以查询修改职工的信息。

总体数据流图

1

住宿

管理

管理员

2

查询管理

3

①住宿数据流图

客房管理

客房号码

客房信息

退宿信息

住宿信息

客房信息

D3 客房信息

D2 住宿信息

D1 退房信息

住宿信息

图2.2总数据流图

图2.5客房数据流图

D6 房客信息

D5 客房信息

管理员

3.1

客房登记

顾客

3.2

客房

管理

住宿信息

退房信息

住宿信息

查询信息

客房信息

预定信息

住宿对象

客房号码

住宿信息

③客房数据流图

图2.4退房数据流图

图2.3住宿数据流图

住宿信息

客房信息

D2 退宿信息

D1 住宿信息

退宿信息

住宿信息

D4 退宿信息

D3 客房信息

客房号码

客房信息

管理员

2.1

查询

登记

顾客

2.2

预定

登记

管理员

1.1

住宿

登记

顾客

1.2

退宿

登记

②退房数据流图

## 2.4 数据需求

### 2.4.1 数据项分析汇总

|  |  |  |  |
| --- | --- | --- | --- |
| 数据项编号 | 数据项名 | 数据项含义 | 存储长度 |
| DI-1 | 房间号 | 房间的号码 | int(6) |
| DI-2 | 入住时间 | 入住房间的时间 | datetime |
| DI-3 | 姓名 | 客人姓名 | char(8) |
| DI-4 | 性别 | 客人的性别 | char(2) |
| DI-5 | 身份证 | 客人身份证号 | char(18) |
| DI-6 | 电话 | 客人电话 | char(11) |
| DI-7 | 类型 | 房间类型 | char(6) |
| DI-8 | 等级 | 房间等级 | char(6) |
| DI-9 | 价格 | 房间价格 | int(6) |
| DI-10 | 状态 | 房间状态 | char(2) |
| DI-11 | 职工号 | 职工的编号 | int(5) |
| DI-12 | 账户 | 职工账户 | char(10) |
| DI-13 | 密码 | 职工密码 | char(50) |
| DI-14 | 职位 | 职工的职位 | char(10) |
| DI-15 | 电话 | 职工电话 | char(11) |
| DI-16 | 住址 | 职工住址 | char(100) |

### 2.4.2 数据整合与分类

|  |  |  |  |
| --- | --- | --- | --- |
| 数据结构编号 | 数据结构名 | 数据结构含义 | 组成 |
| DS-1 | Checkroom | 入住信息 | 房间号，入住时间 |
| DS-2 | guest | 客人信息 | 房间号，姓名，性别，身份证，电话 |
| DS-3 | Room | 客房信息 | 房间号，类型，等级，价格，状态 |
| DS-4 | Workers | 工作人员 | 职工号，账户，密码，职位，电话，住址 |

### 2.4.3 数据字典

数据字典是关于数据信息的集合，即对数据流图中包含的所有元素的定义的集合。

(1)数据结构

|  |  |
| --- | --- |
| 数据结构名 | 属性 |
| Checkroom | 房间号，入住时间 |
| guest | 房间号，姓名，性别，身份证，电话 |
| Room | 房间号，类型，等级，价格，状态 |
| Workers | 职工号，账户，密码，职位，电话，住址 |

(2)数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 数据流名 | 说明 | 数据流来源 | 数据流去向 |
| 住宿信息 | 住宿的信息 | 管理员 | 住宿管理 |
| 客房号码 | 客房标识号码 | 管理员 | 查询管理 |
| 客房信息 | 客房的详细信息 | 管理员 | 客房管理 |
| 退房信息 | 消费信息以及客人信息 | 退宿登记 | 退宿管理 |
| 住宿对象 | 客户的详细信息 | 客房管理 | 房客信息 |

(3)数据项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名及表中列名 | 含义说明 | 数据类型 | 长度 |
| 1 | 房间号 | 房间的号码 | int | 6 |
| 2 | 入住时间 | 入住房间的时间 | datetime |  |
| 3 | 姓名 | 客人姓名 | char | 8 |
| 4 | 性别 | 客人的性别 | char | 2 |
| 5 | 身份证 | 客人身份证号 | char | 18 |
| 6 | 电话 | 客人电话 | char | 11 |
| 7 | 类型 | 房间类型 | char | 6 |
| 8 | 等级 | 房间等级 | char | 6 |
| 9 | 价格 | 房间价格 | int | 6 |
| 10 | 状态 | 房间状态 | char | 2 |
| 11 | 职工号 | 职工的编号 | int | 5 |
| 12 | 账户 | 职工账户 | char | 10 |
| 13 | 密码 | 职工密码 | char | 50 |
| 14 | 职位 | 职工的职位 | char | 10 |
| 15 | 电话 | 职工电话 | char | 11 |
| 16 | 住址 | 职工住址 | char | 100 |

(4)数据存储

|  |  |  |
| --- | --- | --- |
| 数据存储名 | 输入的数据流 | 输出的数据流 |
| 存储客户信息 | 增加或修改的客户信息 | 改变后的客户信息 |
| 存储客房信息 | 增加的客房信息 | 改变后的客房信息 |
| 存储职工信息 | 增加或修改的职工信息 | 改变后的职工信息 |
| 存储入住信息 | 增加入住的信息 | 改变后的入住信息 |

(5)数据处理

|  |  |  |  |
| --- | --- | --- | --- |
| 处理过程名 | 处理说明 | 输入数据流 | 输出数据流 |
| 增加客房信息 | 新的各数据项符合要求 | 要增加的客房信息 | 新增的客房信息 |
| 删除客房信息 | 只删除已有客房信息 | 需要删除的客房号码 | 删除后的客房信息 |
| 增加入住信息 | 新的各数据项符合要求 | 要添加的入住信息以及客户信息 | 添加后的入住信息以及客户信息 |
| 结算客房信息 | 只删除客户信息以及客房状态 | 客房号码 | 结算后的客房信息以及客户信息 |
| 修改客户信息 | 修改不正确的用户信息 | 正确的客户信息 | 更正后的客户信息 |
| 增加职工信息 | 新的各数据项符合要求 | 要添加的职工信息 | 添加后的职工信息 |
| 删除职工信息 | 只删除已有的职工信息 | 要删除的职工号 | 删除后的职工信息 |
| 修改职工信息 | 只能经理才能修改职工信息 | 要修改的职工号 | 修改后的职工信息 |

# 3数据库概念结构设计

概念性信息模型是一种面向问题的数据模型，是按照用户的观点来对数据和信息建模。它描述了从用户角度看到的数据，它反映了用户的现实环境。最常用的表示概念性数据模型方法，是实体——联系方法（Entity-Relationship Approach）。这种方法用E——R图描述现实世界中的实体，而不涉及这些实体在系统中的实现方法。用这种方法表示的概念性数据模型又称为ER模型。ER模型包含“实体”、“联系”、“属性”等三个基本成份。

(1)顾客属性如图3.1

客人信息表

姓名

性别

电话

房间号

身份证号

图3.1客人信息属性图

(2)客房属性如图3.2

房间基本信息表

房间号

房间等级

价格

房间状态

房间类型

图3.2 房间信息属性图

(3)服务员属性如图3.3

职工基本信息

职工号

职工账号

住址

职工密码

电话

职工职位

图3.3职工信息属性图

(4)入住信息属性如图3.4

入住信息

房间号

入住时间

图3.4 入住信息属性图

(5)总E-R图

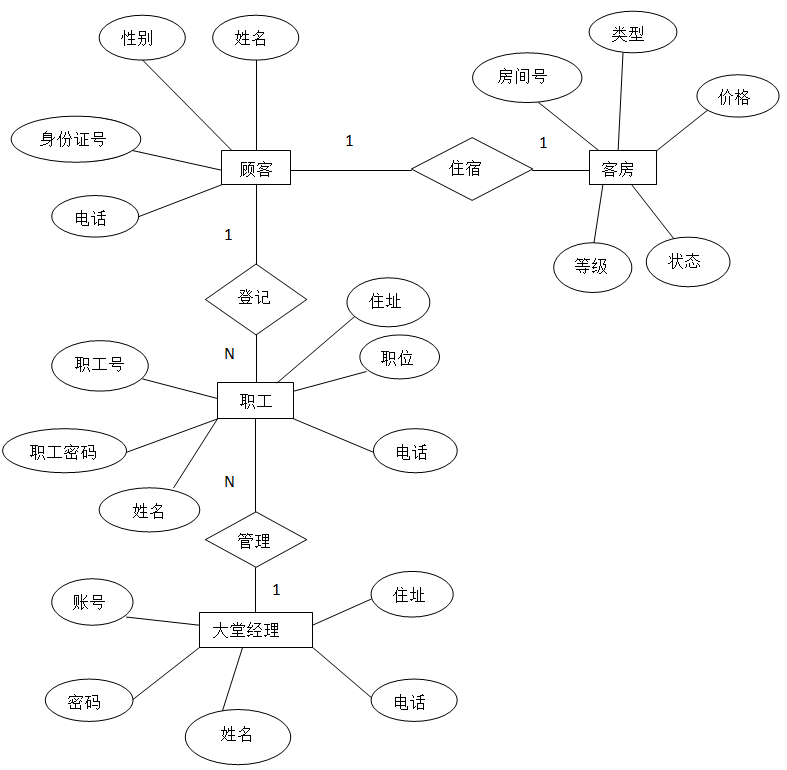


图3.5 总E-R图

# 4 数据库逻辑结构设计

## 4.1实体所对应的关系模式

checkroom（房间号，入住时间）

guest（房间号，姓名，性别，身份证，电话）

room（房间号，类型，等级，价格，状态）

workers（职工号，账户，密码，职位，电话，住址）

## 4.2关系模式优化处理

checkroom（房间号，入住时间）BCNF;

guest（房间号，姓名，性别，身份证，电话）BCNF;

room（房间号，类型，等级，价格，状态）BCNF;

workers（职工号，账户，密码，职位，电话，住址）BCNF;

## 4.3 数据表设计

表4-1 checkroom表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 能否为空 |
| 房间号 | int(6) | NOT NULL |
| 入住时间 | datetime | NOT NULL |

表4-2 guest表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 能否为空 |
| 房间号 | int(11) | NOT NULL |
| 姓名 | char(8) | NOT NULL |
| 性别 | char(2) | NOT NULL |
| 身份证 | char(18) | NOT NULL |
| 电话 | char(11) | NOT NULL |

表4-3 room表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 能否为空 |
| 房间号 | int(11) | NOT NULL |
| 类型 | char(6) | NOT NULL |
| 等级 | char(6) | NOT NULL |
| 价格 | int(6) | NOT NULL |
| 状态 | char(2) | NOT NULL |

表4-4 workers表

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 能否为空 |
| 职工号 | int(5) | NOT NULL |
| 账户 | char(10) | NOT NULL |
| 密码 | char(50) | NOT NULL |
| 职位 | char(10) | NOT NULL |
| 电话 | char(11) | NOT NULL |
| 住址 | char(100) | NOT NULL |

# 5 数据库物理结构设计

## 5.1 系统采用的数据库简介

MySQL 是一个关系型数据库，由瑞典 MySQL AB 公司开发。MySQL 最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策(本词条"授权政策")，它分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。由于其社区版的性能卓越，搭配 PHP ,Linux和 Apache 可组成良好的开发环境，经过多年的web技术发展,在业内被广泛使用的一种web服务器解决方案之一,称之为LAMP。

## 5.2 数据库的建立和结构设计

表checkroom的建立:

CREATE TABLE IF NOT EXISTS `checkroom` (

`房间号` int(6) NOT NULL,

`入住时间` datetime NOT NULL,

PRIMARY KEY (`房间号`),

UNIQUE KEY `房间号` (`房间号`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

表guest的建立:

CREATE TABLE IF NOT EXISTS `guest` (

`房间号` int(11) NOT NULL,

`姓名` char(8) COLLATE utf8\_bin NOT NULL,

`性别` char(2) COLLATE utf8\_bin NOT NULL,

`身份证` char(18) COLLATE utf8\_bin NOT NULL,

`电话` char(11) COLLATE utf8\_bin NOT NULL,

PRIMARY KEY (`房间号`),

UNIQUE KEY `房间号` (`房间号`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

表room的建立:

CREATE TABLE IF NOT EXISTS `room` (

`房间号` int(11) NOT NULL,

`类型` char(6) COLLATE utf8\_bin NOT NULL,

`等级` char(6) COLLATE utf8\_bin NOT NULL,

`价格` int(6) NOT NULL,

`状态` char(2) COLLATE utf8\_bin NOT NULL DEFAULT 'N',

PRIMARY KEY (`房间号`),

UNIQUE KEY `房间号` (`房间号`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

表workers的建立:

CREATE TABLE IF NOT EXISTS `workers` (

`职工号` int(5) NOT NULL,

`账户` char(10) COLLATE utf8\_bin NOT NULL,

`密码` char(50) COLLATE utf8\_bin NOT NULL,

`职位` char(10) COLLATE utf8\_bin NOT NULL,

`电话` char(11) COLLATE utf8\_bin NOT NULL,

`住址` char(100) COLLATE utf8\_bin NOT NULL,

PRIMARY KEY (`职工号`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

## 5.3 数据库的完整性设计

表checkroom:

CREATE TABLE IF NOT EXISTS `checkroom` (

`房间号` int(6) NOT NULL, /\*用户自定义完整性\*/

`入住时间` datetime NOT NULL, /\*用户自定义完整性\*/

PRIMARY KEY (`房间号`), /\*实体完整性\*/

UNIQUE KEY `房间号` (`房间号`) /\*用户自定义完整性\*/

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

表guest:

CREATE TABLE IF NOT EXISTS `guest` (

`房间号` int(11) NOT NULL, /\*用户自定义完整性\*/

`姓名` char(8) COLLATE utf8\_bin NOT NULL, /\*用户自定义完整性\*/

`性别` char(2) COLLATE utf8\_bin NOT NULL, /\*用户自定义完整性\*/

`身份证` char(18) COLLATE utf8\_bin NOT NULL, /\*用户自定义完整性\*/

`电话` char(11) COLLATE utf8\_bin NOT NULL, /\*用户自定义完整性\*/

PRIMARY KEY (`房间号`), /\*实体完整性\*/

UNIQUE KEY `房间号` (`房间号`) /\*用户自定义完整性\*/

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

表room:

CREATE TABLE IF NOT EXISTS `room` (

`房间号` int(11) NOT NULL, /\*用户自定义完整性\*/

`类型` char(6) COLLATE utf8\_bin NOT NULL, /\*用户自定义完整性\*/

`等级` char(6) COLLATE utf8\_bin NOT NULL, /\*用户自定义完整性\*/

`价格` int(6) NOT NULL, /\*用户自定义完整性\*/

`状态` char(2) COLLATE utf8\_bin NOT NULL DEFAULT 'N',/\*用户自定义完整性\*/

PRIMARY KEY (`房间号`), /\*实体完整性\*/

UNIQUE KEY `房间号` (`房间号`) /\*用户自定义完整性\*/

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

表workers:

CREATE TABLE IF NOT EXISTS `workers` (

`职工号` int(5) NOT NULL,/\*用户自定义完整性\*/

`账户` char(10) COLLATE utf8\_bin NOT NULL,/\*用户自定义完整性\*/

`密码` char(50) COLLATE utf8\_bin NOT NULL,/\*用户自定义完整性\*/

`职位` char(10) COLLATE utf8\_bin NOT NULL,/\*用户自定义完整性\*/

`电话` char(11) COLLATE utf8\_bin NOT NULL,/\*用户自定义完整性\*/

`住址` char(100) COLLATE utf8\_bin NOT NULL,/\*用户自定义完整性\*/

PRIMARY KEY (`职工号`) /\*实体完整性\*/

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8\_bin;

## 5.4存储方法设计

存储时，我们不单单只是输入一个表单，存储一个表，而是输入一个表单，每个表选择自己需要的信息进行存储，这就牵扯到数据的筛选，首先是判断，我们数据库中的表不为空表，所以每个数据都必须有效且不为空，在基本的判断之后我们就不用后端语言进行判断哪个表需要哪个内容了，因为那样会增加后端的运行时间还会导致多重循环带来的不必要的对数据库的多次访问，所以我们将这项筛选工作交给数据库自己去实现，这样只连接一次数据库就可以了，在insert into 表名 后面添加我们需要的信息，数据库就可以根据我们设置的值自动存入自己需要的值，这样就减少了后台服务器的工作，增加了网页的反应速度。

## 5.5存取结构设计

为了方便存取，我们将需要连接查询的表都设置了一个主键”房间号”，在需要查询多个表的数据并进行处理是我们可以用select \* from room,checkroom,guest where room.房间号=guest.房间号 AND guest.房间号=checkroom.房间号 order by room.房间号 asc这种方法，将需要的信息查询并连接成一个表，这个就有利于我们在前端现实的时候方便查找。

## 5.6查询优化

可以通过如下方法来优化查询:

(1)尽量避免在 where 子句中使用!=或<>操作符，否则将引擎放弃使用索引而进行全表扫描。

(2)尽量避免在 where 子句中对字段进行 null 值判断，否则将导致引擎放弃使用索引而进行全表扫描，如：select id from t where num is null可以在num上设置默认值0，确保表中num列没有null值，然后这样查询：select id from t where num=0。

(3)尽量避免在 where 子句中使用 or 来连接条件，否则将导致引擎放弃使用索引而进行全表扫描，如：select id from t where num=10 or num=20可以这样查询：select id from t where num=10 union all select id from t where num=20。

(4)如果在 where 子句中使用参数，也会导致全表扫描。因为SQL只有在运行时才会解析局部变量，但优化程序不能将访问计划的选择推迟到运行时;它必须在编译时进行选择。然而，如果在编译时建立访问计划，变量的值还是未知的，因而无法作为索引选择的输入项。如下面语句将进行全表扫描：select id from t where num=@num 可以改为强制查询使用索引：select id from t with(index(索引名)) where num=@num。

(5)在使用索引字段作为条件时，如果该索引是复合索引，那么必须使用到该索引中的第一个字段作为条件时才能保证系统使用该索引，否则该索引将不会被使用，并且应尽可能的让字段顺序与索引顺序相一致。

(6)不要在 where 子句中的“=”左边进行函数、算术运算或其他表达式运算，否则系统将可能无法正确使用索引。

# 6 编码实现

## 6.1 系统详细功能

顾客模块

顾客到来

登记信息

入住

消费结算

顾客离开

图6.1 顾客模块

职工模块

选择房间号

录入顾客信息

安排顾客

修改顾客信息

收款

图6.2 职工模块

内部管理模块

添加职工

录入职工信息

修改职工信息

删除职工信息

职工离开

图6.3 内部管理模块

## 6.2 系统开发语言简介

PHP全称英文超级文本预处理语言（Hypertext Preprocessor），它是在服务器端执行的脚本语言，与C语言类似，是常用的网站编程语言。PHP语言主要有以下特点：(1)开源性和免费性。由于PHP的解释器的源代码是公开的，所以安全系数较高的网站可以自己更改PHP的解释程序。另外，PHP运行环境的使用也是免费的。(2)跨平台性强。由于PHP的解释器是开源的，所以能够在所有的操作系统平台上非常稳定地运行，这使它成为常用的服务器语言。(3)快捷性。PHP是一种非常容易学习和使用的一门语言，它的语法特点类似于C语言，但又没有C语言复杂的地址操作，而且又加入了面向对象的概念，再加上它具有简洁的语法规则，使得它操作编辑非常简单，实用性很强。(4)面向过程和面向对象并用。在PHP语言的使用中，可以分别使用面向过程和面向对象，而且可以将PHP面向过程和面向对象两者一起混用，这是其他很多编程语言是做不到的。(5)运行高效性。由于PHP运行在相应的平台解释器上，消耗系统资源比较少，运行的环境简单，所以效率就很高。(6)数据库连接的广泛性。PHP可以与很多主流的数据库建立起连接，如MySQL、ODBC、Oracle、AdabasD、S等，PHP是利用编译的不同函数与这些数据库建立起连接的，PHPLIB就是常用的为一般事务提供的基库。

## 6.3 系统的实现

登录界面代码如下:

<div class="panel panel-default" style="margin-left: 35%;margin-right: 35%;margin-top: 150px;">

<div class="panel-body">

</br>

<p style="margin-left: 32%; font-family: 幼圆; font-size: 18px;">登陆宾馆管理系统</p>

</br>

<form class="form-horizontal" style="margin-left: 5%;margin-right: 5%;" method="post" action="login\_server.PHP">

<div class="form-group">

<label for="" class="col-sm-1 control-label"></label>

<div class="col-sm-10">

<input type="text" class="form-control" id="" placeholder="账号" name="账户">

</div>

</div>

<div class="form-group">

<label for="" class="col-sm-1 control-label"></label>

<div class="col-sm-10">

<input type="password" class="form-control" id="" placeholder="密码" name="密码">

</div>

</div>

<div class="form-group">

<div class="col-sm-offset-2 col-sm-10">

<button type="submit" class="btn btn-default" style="margin-left: 25%;">登录系统</button>

</div>

</div>

</form>

</div>

</div>

登录界面如下:

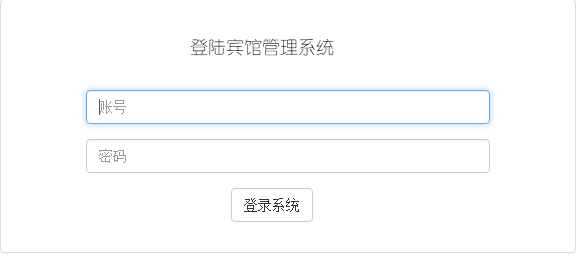
****

图6.4 登录界面

主界面代码如下:

<table class="table table-hover text-center">

<!-- On rows -->

<?PHP

$Z=0;

$NZ=0;

echo"<th class='text-center'>房间号</th>";

echo"<th class='text-center'>房间类型</th>";

echo"<th class='text-center'>房间等级</th>";

echo"<th class='text-center'>价格</th>";

echo"<th class='text-center'>状态</th>";

echo"<th class='text-center'>操作</th>";

while($rows=mysqli\_fetch\_assoc($rs)){

++$Z;

if($rows["状态"]!=='N'){++$NZ; echo "<tr class='danger' >";}else{

echo "<tr class='success' >";

}

echo"<td >".$rows["房间号"]."</td>";

echo"<td >".$rows["类型"]."</td>";

echo"<td >".$rows["等级"]."</td>";

echo"<td >".$rows["价格"]."/天</td>";

echo"<td >".$rows["状态"]."</td>";

if($rows["状态"]=='N'){

echo"<td ><a href='add\_check.PHP?id=".$rows['房间号']."'>入住</a>

</td>";}else{ echo"<td ><a data-target='#myModal' data-toggle='modal' href='Settlement.PHP?id=".$rows['房间号']."'>结算</a></td>";}

echo"</tr>";

}

$LV=($NZ/$Z)\*100;

?>

</table>

<div class="progress">

<div class="progress-bar progress-bar-striped active" role="progressbar" aria-valuenow="<?PHP echo round($LV,2); ?>" aria-valuemin="0" aria-valuemax="100" style="width: <?PHP echo round($LV,2); ?>%">

当前宾馆客房利用率:<?PHP echo round($LV,2); ?>%

</div>

</div>

<button type="button" class="btn btn-default" aria-label="Left Align" onclick="javascrtpt:window.location.href='add\_room.PHP'">

客房添加

</button>

<button type="button" class="btn btn-default" aria-label="Left Align" onclick="javascrtpt:window.location.href='add\_room\_delet.PHP'">

客房删除

</button>

<div class="modal fade bs-example-modal-sm" id="myModal" tabindex="-1" role="dialog" aria-labelledby="myModalLabel">

<div class="modal-dialog modal-sm" role="document">

<div class="modal-content">

</div>

</div>

</div>

主界面图如下:



图6.5 主界面

客房添加代码如下:

<form class="text-center" style="margin-top: 50px;margin-left: 500px;margin-right: 500px;" method="post" action="add\_room\_server.PHP">

<div class="form-group">

<label for="exampleInputEmail1">房间号</label>

<input type="text" class="form-control" id="exampleInputEmail1" name="房间号">

</div>

<div class="form-group">

<label for="exampleInputPassword1">类型</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="类型">

</div>

<div class="form-group">

<label for="exampleInputPassword1">等级</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="等级">

</div>

<div class="form-group">

<label for="exampleInputPassword1">价格</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="价格">

</div>

<button type="submit" class="btn btn-default">提交</button>

</form>

客房添加界面如下:

图6.6 添加客房图

客户入住界面代码如下:

<form class="text-center" style="margin-top: 50px;margin-left: 500px;margin-right: 500px;" method="post" action="add\_check\_server.PHP">

<div class="form-group">

<label for="exampleInputEmail1">房间号</label>

<input type="text" class="form-control" id="exampleInputEmail1" value="<?PHP echo $\_GET["id"]; ?>" name="房间号">

</div>

<div class="form-group">

<label for="exampleInputPassword1">姓名</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="姓名">

</div>

<div class="form-group">

<label for="exampleInputPassword1">性别</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="性别">

</div>

<div class="form-group">

<label for="exampleInputPassword1">电话</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="电话">

</div>

<div class="form-group">

<label for="exampleInputPassword1">身份证</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="身份证">

</div>

<div class="form-group">

<label for="exampleInputPassword1">入住时间</label>

<input type="text" class="form-control" id="exampleInputPassword1" value="<?PHP echo @date("Y-m-d H:i:s", time()+8\*60\*60); ?>" name="入住时间">

</div>

<button type="submit" class="btn btn-default">提交</button>

</form>

客户入住界面如下:****

图6.7 客户入住界面图

客户信息查询代码如下:

<table class="table table-hover text-center">

<!-- On rows -->

<?PHP

echo"<th class='text-center'>房间号</th>";

echo"<th class='text-center'>姓名</th>";

echo"<th class='text-center'>性别</th>";

echo"<th class='text-center'>身份证</th>";

echo"<th class='text-center'>电话</th>";

echo"<th class='text-center'>类型</th>";

echo"<th class='text-center'>等级</th>";

echo"<th class='text-center'>价格</th>";

echo"<th class='text-center'>入住时间</th>";

echo"<th class='text-center'>操作</th>";

while($rows=mysqli\_fetch\_assoc($rs)){

echo "<tr>";

echo"<td >".$rows["房间号"]."</td>";

echo"<td >".$rows["姓名"]."</td>";

echo"<td >".$rows["性别"]."</td>";

echo"<td >".$rows["身份证"]."</td>";

echo"<td >".$rows["电话"]."</td>";

echo"<td >".$rows["类型"]."</td>";

echo"<td >".$rows["等级"]."</td>";

echo"<td >".$rows["价格"]."</td>";

echo"<td >".$rows["入住时间"]."</td>";

echo"<td ><a href='user\_update.PHP?id=".$rows['房间号']."&姓名=".$rows["姓名"]."&性别=".$rows["性别"]."&电话=".$rows["电话"]."&身份证=".$rows["身份证"]."&入住时间=".$rows["入住时间"]."'>信息修改</a></td>";

echo"</tr>";

}

?>

</table>

客户信息查询界面如下:

图6.8 客户信息查询图

客户信息修改界面代码如下:

<form class="text-center" style="margin-top: 50px;margin-left: 500px;margin-right: 500px;" method="post" action="user\_update\_server.PHP">

<div class="form-group">

<label for="exampleInputEmail1">房间号</label>

<input type="text" class="form-control" id="exampleInputEmail1" value="<?PHP echo $\_GET["id"];?>" name="房间号">

</div>

<div class="form-group">

<label for="exampleInputPassword1" >姓名</label>

<input type="text" class="form-control" id="exampleInputPassword1" value="<?PHP echo $\_GET["姓名"]; ?>" name="姓名">

</div>

<div class="form-group">

<label for="exampleInputPassword1">性别</label>

<input type="text" class="form-control" id="exampleInputPassword1" value="<?PHP echo $\_GET["性别"]; ?>" name="性别">

</div>

<div class="form-group">

<label for="exampleInputPassword1">电话</label>

<input type="text" class="form-control" id="exampleInputPassword1" value="<?PHP echo $\_GET["电话"]; ?>" name="电话">

</div>

<div class="form-group">

<label for="exampleInputPassword1">身份证</label>

<input type="text" class="form-control" id="exampleInputPassword1" value="<?PHP echo $\_GET["身份证"]; ?>" name="身份证">

</div>

<div class="form-group">

<label for="exampleInputPassword1">入住时间</label>

<input type="text" class="form-control" id="exampleInputPassword1" value="<?PHP echo $\_GET["入住时间"]; ?>" name="入住时间">

</div>

<button type="submit" class="btn btn-default">提交</button>

客户信息修改界面:

****

图6.9 客户信息修改界面

职工添加界面代码如下:

<form class="text-center" style="margin-top: 50px;margin-left: 500px;margin-right: 500px;" method="post" action="add\_work\_server.PHP">

<div class="form-group">

<label for="exampleInputEmail1">职工号</label>

<input type="text" class="form-control" id="exampleInputEmail1" name="职工号">

</div>

<div class="form-group">

<label for="exampleInputPassword1">账户</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="账户">

</div>

<div class="form-group">

<label for="exampleInputPassword1">密码</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="密码">

</div>

<div class="form-group">

<label for="exampleInputPassword1">职位</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="职位">

</div>

<div class="form-group">

<label for="exampleInputPassword1">电话</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="电话">

</div>

<div class="form-group">

<label for="exampleInputPassword1">住址</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="住址">

</div>

<button type="submit" class="btn btn-default">提交</button>

</form>

职工添加界面:

****

图6.10 职工添加界面

职工信息查询代码如下:

<table class="table table-hover text-center">

<!-- On rows -->

<?PHP

echo"<th class='text-center'>职工号</th>";

echo"<th class='text-center'>账户</th>";

echo"<th class='text-center'>职位</th>";

echo"<th class='text-center'>电话</th>";

echo"<th class='text-center'>住址</th>";

echo"<th class='text-center'>操作</th>";

while($rows=mysqli\_fetch\_assoc($rs)){

echo "<tr>";

echo"<td >".$rows["职工号"]."</td>";

echo"<td >".$rows["账户"]."</td>";

echo"<td >".$rows["职位"]."</td>";

echo"<td >".$rows["电话"]."</td>";

echo"<td >".$rows["住址"]."</td>";

if($\_SESSION["职位"]=="大堂经理"){

echo"<td ><a href='update\_work.PHP?id=".$rows['职工号']."&账户=".$rows["账户"]."&职位=".$rows["职位"]."&电话=".$rows["电话"]."&住址=".$rows["住址"]."&密码=".$rows["密码"]."'>修改</a>";if($rows["职位"]=="大堂经理"){}else{echo "<a href='user\_delet\_server.PHP?id=".$rows['职工号']."'>删除</a>";}

echo "</td>";}else{ echo"<td >没有权限</td>";}

echo"</tr>";

}

?>

</table>

<button type="button" class="btn btn-default" onclick="javascrtpt:window.location.href='add\_work.PHP'">

添加职工

</button>

职工信息查询界面如下:

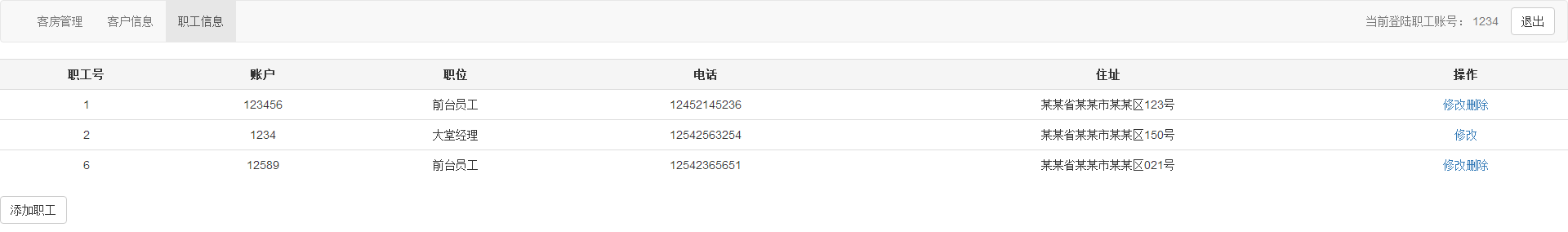
****

图6.11 职工信息查询界面

结算信息查询代码如下:

while ($rows=mysqli\_fetch\_assoc($rs)){

if ($rows["房间号"]==$\_GET["id"]){

$F=$rows["房间号"];

$X=$rows["姓名"];

$XB=$rows["性别"];

$S=$rows["身份证"];

$D=$rows["电话"];

$L=$rows["类型"];

$DJ=$rows["等级"];

$JG=$rows["价格"];

$RZ=$rows["入住时间"];

}

}

$T=@date("Y-m-d H:i:s",time());

$T1=@strtotime($T);

$T2=@strtotime($RZ);

$T3=ceil(($T1-$T2)/86400);

?>

<ul class="list-group">

<li class="list-group-item">房间号：<?PHP echo $F;?></li>

<li class="list-group-item">姓名：<?PHP echo $X; ?></li>

<li class="list-group-item">性别：<?PHP echo $XB; ?></li>

<li class="list-group-item">身份证：<?PHP echo $S; ?></li>

<li class="list-group-item">电话：<?PHP echo $D; ?></li>

<li class="list-group-item">类型：<?PHP echo $L; ?></li>

<li class="list-group-item">等级：<?PHP echo $DJ; ?></li>

<li class="list-group-item">价格：<?PHP if($T3==0){echo $JG;}else{echo $JG\*$T3;} ?></li>

<li class="list-group-item">入住时间：<?PHP echo $RZ; ?></li>

</ul>

<div class="btn-group text-center" style="margin-left: 37%;margin-bottom: 5%;" role="group" aria-label="...">

<button type="button" class="btn btn-default" onclick="javascrtpt:window.location.href='over.PHP?id=<?PHP echo $F;?>'">结算</button>

</div>

结算信息查询界面如下:

****

图6.12 结算界面

## 6.4 数据库连接代码及数据操作代码

数据库连接代码:

<?PHP

$db\_link=mysqli\_connect("localhost", //MYSQL服务器地址

"root", //MYSQL登录用户名

"root", //MYSQL登录密码（此处可根据实际情况修改）

"ks"); //需要连接数据库

if(!$db\_link)

{

printf("Can't connect to MySQL Server.Errorcode:%s ", mysqli\_connect\_error($db\_link));

exit;

}

//设置数据库编码

mysqli\_query($db\_link,"SET NAMES 'utf8'");

session\_start();

?>

数据库插入代码:

$sql1 = "insert into room (房间号,类型,等级,价格) values('".$\_POST["房间号"]."','".$\_POST["类型"]."','".$\_POST["等级"]."','".$\_POST["价格"]."')";

mysqli\_query($db\_link,$sql1) or die ("在room表中插入记录失败：".mysqli\_error());

数据库删除代码:

$sql1 = "delete from room where 房间号='".$\_POST["房间号"]."'";

mysqli\_query($db\_link,$sql1) or die ("在room表中插入记录失败：".mysqli\_error());

数据库更新代码:

$sql1 = "update workers set 账户='".$\_POST['账户']."',密码='".$\_POST['密码']."',职位='".$\_POST['职位']."',电话='".$\_POST['电话']."',住址='".$\_POST['住址']."' where 职工号='".$\_POST["职工号"]."'";

mysqli\_query($db\_link,$sql1) or die ("在workers表中更新记录失败：".mysqli\_error($db\_link));

数据库查询代码:

$sql = "select \* from room,checkroom,guest where room.房间号=guest.房间号 AND guest.房间号=checkroom.房间号";

$rs=mysqli\_query($db\_link,$sql);

$rows=mysqli\_fetch\_assoc($rs);

# 7 测试分析报告

这次我选做的课程设计题目是宾馆住宿系统，题目要求该软件能够解决日常生活中有关宾馆的一系列问题，可以完成对到来的顾客登记，安排住宿情况，离开结算等问题。程序中从顾客到来，登记顾客信息，到安排顾客住宿都有相应的功能去实现，顾客可以选择房间号，房间等级，系统会存储客人信息，在后来可以根据客人的需求更换房间，因此也可以更改客人入住信息，满足顾客临时的需求。直到顾客离开，可以查询到顾客的住宿情况，自动根据入住天数，还有房间等级，消费情况，结算数据，让宾馆住宿办理手续变得简单方便。除此之外该软件也能用于宾馆内部人员的管理，在职工入职的时候会分配职工号和职工密码，方便他们管理，如职工有更改，系统也可以修改职工信息，更高级的权限还能删除职工账号，便于上级管理。软件的后台我们用数据库来容纳所有的信息，创建所需要的表，然后和软件连接起来，实现该系统的功能。

在这次设计中也出现了很多问题，数据库的连接不能很熟练，导致连接起来浪费很多时间，在软件完成后，界面比较简单，视觉效果不好，就是用来完成一些日常的操作，因为时间的关系，所以很多深层次的功能都没有涉及到，要用到现实生活中还需要更多的去添加一些深层次的功能。

# 总结

在这次课程设计中，我选用了PHP以及MySQL作为课设的主要语言，在使用的过程中发现很多函数的用法以及参数都和我学的变了很多，所以在写代码的时候碰到了很多错误，但是经过查找问题的解决办法以及官方的帮助文档都得到了解决。

在设计中，我们采用了phpMyAdmin作为管理数据库的工具，这主要是我们用的是网页语言，在管理数据库时用网页可视化界面管理MySQL比较的方便，这大大提高了写程序的速度，而且phpMyAdmin是我们安装PHP的运行环境时自带的一个MySQL管理软件，我们就不用在单独安装MySQL以及他的前端管理界面。

刚开始写PHP的时候想的是用HTML5作为前端界面，PHP只作为后端处理语言，但是在HTML5与PHP的传值中发现这需要一个很大的框架作为基础才能实现，单独的话只能使用JS进行传值，但是JS的传值远远没有PHP内部的传值快，所以我们就采用了将HTML5代码嵌套在PHP文件中进行对前端的显示。

由于时间的限制，程序中还有一些安全性的问题没有解决，但是基础功能全部实现了，还进行了对前端的美化设计，程序中的这些问题也就是该程序的缺陷了吧。

# 参考文献

[1] 王珊.数据库系统概论（第四版）.高等教育出版社.2014.12

[2] 刘金岭.数据库原理及应用实验与课程设计指导.清华大学出版社出版.2010.6

[3] 钱雪忠.数据库原理及技术课程设计(全新正版）.清华大学出版社出版.2009.2

[4] 周屹.数据库原理及开发应用－实验与课程设计指导. 清华大学出版社出版.2008.12

[5] 李春葆.Visual Basic数据库系统设计与开发.清华大学出版社出版.2003.8

[6] 唐红亮.SQL Server数据库设计与系统开发教程.清华大学出版社出版.2007.10

[7] 丁忠俊.数据库系统原理及应用.清华大学出版社出版.2012.4

[8] 李晓峰.数据库系统原理及应用.水利水电出版社.2011.2

[9] 万常选.数据库系统原理与设计(第2版).清华大学出版社出版.2012.9

[10] 冯健华.数据库系统设计与原理（第2版）.清华大学出版社出版.2012.8

[11]（美）罗摩克里希纳，格尔基.周立栋译.数据库管理系统原理与设计(第3版)，清华大学出版社出版.2004.3

[12]（美）西尔伯沙茨.杨冬青译.数据库系统概念（原书第6版•本科教学版）.机械工业出版社.2013.1

[13] 何先军.Access 2010数据库应用从入门到精通.中国铁道出版社.2013.10

[14] 辛赫（Singh.S.K）[印].何玉洁,王晓波，车蕾等译.数据库系统概念、设计及应用.机械工业出版社2010.1

# 致谢

我认为，在这学期的课程设计中，在收获知识的同时，还收获了阅历，收获了成熟，在此过程中，我们通过查找大量资料，请教老师和同学。使我再专业知识和动手实践方面都得了到很好的提升，在此，要对给过我帮助的所有同学和各位指导老师再次表示忠心的感谢!课设的成功，少不了老师的耐心指导和同学的热心帮助，以及小组中其他成员的大力配合。没有大家的默契，也收获不了今日的成功，在课设的过程中每一个人都努力查找资料，仔细检查，认真核对，都付出了自己的努力和艰辛，在此，谢谢所有人的努力和帮助，才使实验能够得以顺利地完成。

　　由于本人的设计能力有限，在设计过程中难免出现错误，恳请老师们多多指教,我十分乐意接受你们的批评与指正，本人将万分感谢