

## 目录

摘要 .....	1
Abstract .....	2
第一章 绪论 .....	4
1.1 课题背景 .....	4
1.2 国内外现状与趋势 .....	5
1.3 课题内容 .....	6
1.4 论文组织结构 .....	6
第二章 相关技术与方法介绍 .....	7
2.1 相关技术介绍 .....	7
2.2 系统开发方法概述 .....	9
2.3 本章小结 .....	9
第三章 系统分析 .....	10
3.1 系统需求概述 .....	10
3.2 业务流程调查与分析 .....	10
3.3 系统逻辑模型 .....	11
3.4 本章小结 .....	15
第四章 系统设计 .....	16
4.1 系统设计概述 .....	16
4.2 系统总体设计 .....	17
4.3 功能模块详细设计 .....	17
4.4 数据库设计 .....	22
4.5 本章小结 .....	27
第五章 系统实现 .....	28
5.1 项目系统层次设计 .....	28
5.2 各层的具体实现 .....	29
5.3 系统核心功能模块实现 .....	31
5.4 本章小结 .....	45
第六章 系统测试 .....	46

6.1 系统测试的目的及意义 .....	46
6.2 系统功能性测试 .....	46
6.2.1 普通用户注册测试 .....	46
6.2.2 普通用户登录测试 .....	47
6.2.3 会议室预约测试 .....	48
6.2.4 密码修改测试 .....	49
6.2.5 在线留言测试 .....	50
6.3 本章小结 .....	50
第七章 总结 .....	52
参考文献 .....	53
致 谢 .....	55

## 摘要

近些年来，伴随计算机技术和网络技术的快速发展与应用的不断深入，各行各业几乎都在依赖着计算机网络技术开展多种业务活动，企业信息化的水平也在不断提高。伴随企业业务的增多，在企业部门中需要频繁地开展各种会议，而传统的会议室预约采用人工管理方式大大降低了企业部门的办公效率，已经逐渐无法满足企业部门对会议室预约管理的实际需求。企业部门亟需利用现代信息化、计算机网络技术来构建一套会议室预约系统，以实现会议室预约的高效性和简便性，提高企业会议室预约管理的科学化和规范化。因此本文提出了基于 Java 和 JSP 的楼宇会议室预约系统的设计，主要工作内容如下。

首先，对系统中使用的 MVC 框架、B/S 架构、JSP 等相关技术进行了基本介绍，并简要概述了这些技术的特点和优点用途。使用结构化分析方法对系统进行需求分析、业务流程调查与分析操作，然后使用数据流程图和数据字典来分析系统逻辑模型，为后续系统的设计与实现打下了基础。其次，通过前期的分析过程，对系统的整体结构、应有的功能模块、数据库进行设计，并作出各个功能模块的流程图，作出数据库设计过程的总体 E-R 图和应具有的数据表。然后，详细说明了楼宇会议室预约系统主要核心功能模块的实现过程以及代码，给出了实现功能模块界面的截图和核心代码的截图。最后，对系统的功能模块进行功能测试，简要描述了各功能测试的过程，并对测试的方案和结果记录在表格中，与系统实际效果图进行展示。

**关键词：**会议室；预约系统；MVC；B/S

## **Abstract**

In recent years, with the rapid development of computer technology and network technology and the application of the continuous development, all walks of life seem to rely on computer and network technology to conduct business and activities, the level of enterprise information technology continues to improve. With the increase of enterprise management, enterprise departments need to often hold a variety of meetings, and the traditional conference room appointment using artificial management mode greatly reduces the office efficiency of the enterprise department, has gradually been unable to meet the actual needs of the enterprise department on the room appointment management. It is urgent for enterprise departments to construct conference room reservation system by using modern information technology and computer network technology to realize the efficiency and simplicity of conference room reservation and improve the scientific and standardized management of conference room reservation in enterprise departments. Therefore, this article describes the design of java-based and jsp-based building meeting room reservation system, the main work is as follows.

First, the MVC framework, B/S architecture, JSP and other related technologies used in the system are introduced, and the characteristics and advantages of these technologies are briefly introduced. The analysis of system requirements, business process investigation and analysis by structured system analysis method, and data flow chart and data dictionary analysis system logic model are used to lay the foundation for the design and implementation of the subsequent system. Secondly, through the early analysis process, the overall structure of the system, functional modules, database design, and make a flow chart of each functional module, and the database design process of the overall E-R diagram, as well as the data table. Then, the implementation process of the main core function module of the building conference room reservation system is introduced in detail, and the interface and core code screenshot of the implementation function module are given. At last, functional testing of the system, the operation of each function test is briefly described, and the scheme and results of the test are recorded in the table to display the actual effect of the system.

**Keyword:** conference room; reservation system; MVC; B/S

Written by: Li Shu-ying

Supervised by: Ling Xing-hong

# 第一章 绪论

## 1.1 课题背景

会议室是企业单位处理日常业务的重要场所，是一种共享资源，但随着企业事务规模扩大，会议室资源有限，有的企业直接将多个会议室分配给不同的部门使用，而不进行统一管理调配，导致出现有的部门会议室利用率很低，而有的部门却不够用的情况，因而出现资源大量浪费的现象。传统的会议室管理方式，一般都是采用人工管理并结合纸质记录的方式来管理会议室预约，此过程既繁琐又易产生冲突，缺乏灵活的调换机制。用户在查询会议室时，不能实时了解这个会议室的当前预约情况，通常都是需要经历预约咨询、填送报表、签字盖章等过程[2]，因此出现体验不佳，效率低下等弊端。且采用传统的人工管理方式，管理员在面对大量的会议室资源和会议室预约申请的情况下，无法做到严格审查对于相同会议室的不同预约申请时间是否有重叠，且多个部门在同一个时间段来占用同一个会议室的情况也很有可能发生[5]，因此导致出现了会议室预约冲突和资源利用不充分等问题。在发生预约冲突后，也无法及时准确的反映会议室使用情况，查找已被占用和未被占用的会议室比较困难，无法及时的协调，在很大程度上影响了部门的办公效率。

为了使会议室的预约管理和使用进一步实现规范化，能实时地为用户显示会议室的占用信息，避免资源的冲突，提高预约效率，提供灵活高效调换机制，亟需建立楼宇会议室预约系统。会议室预约系统是一个适合企业单位内各个组织部门进行会议室资源管理和使用的系统，可把它部署在企业内部网络中，它主要的功能有会议室预约、预约修改、预约审核等，可以节省大量的时间和人力，同时设置了可以预约的时间段，避免不同部门开会时间的碰撞，极大的利用了会议室资源，且能及时的进行冲突协调，能够满足一般企业会议室预约的需要。该系统中的用户只需通过成功注册账号登录系统，即可便捷地查找了解每个会议室的目前状态和每个时段的预约情况等，并且同时设置了管理员，可以对用户进行限制以及预约的审核，简化了预约流程，节省了管理员的管理时间和精力，摒弃了传统过程中的纸质记录，在很大程度上改善了传统预约方式的弊端。从另一个角度来说，它可以提高企事业单位信息自动化的办公水平，提高日常工作的效益和效率，是当今信息化进程中为提高企业办公效率的一个理想的解决平台[5]。

## 1.2 国内外现状与趋势

美国是最早利用信息管理系统进行会议室预约管理的国家，随着技术的发展，基于 C/S 架构的会议室预约管理系统相继出现，如 CyberMatrix Meeting Manager、Meeting Room Manager[8]等，这些系统可实现固定设备查看会议室预约情况，实现较灵活的会议室调度功能。在国外，也陆续出现了一些高效的视频会议系统，主要是将会议室预约化为线上嵌入其中，例如 ZOOM、Adobe 等视频会议软件。人工智能的不断发展与应用，国外陆续出现了一些具备智能化的会议室预约系统，例如微软中心的会议室，采用全智能模式，语音识别等技术，让预约管理显得更加高效。

我国的计算机管理技术起步稍晚，且大部分企业长期保持着传统的预约管理模式。通过技术的引入，并结合我国企事业单位的办公性质，经过几年的研发和推广，我国的信息管理系统取得了迅猛的发展。各行各业都开始引入计算机技术进行会议室预约管理，改革了人工管理的形式，使得各企事业单位的办公效率得到大幅提升，同时也将使得越来越多符合我国办公模式的技术相继出现。近几年，我国大部分企业都引入了 OA 系统预定会议室[4]，企业单位应用 OA 系统可以达到信息资源的共享目的，从而实现高效的协作工作和管理。然而，会议室预约系统只是 OA 系统的一个附加模块，对于会议室预约管理工作缺乏一定的针对性，相关的管理人员没有办法对会议室采取远程管理，因而 OA 系统只能实现部分会议室预约功能，无法满足用户个性化的需求。但目前，我国也有了一些相对成熟的楼宇会议室预约系统，且有些已进入商业化模式运作，比较出色的如 DMRS 会议室预约系统、iComi 智能会议管理系统、北京科佰思特信息技术有限公司的会议室预约系统等，这些系统都随着 C/S 架构发展为 B/S 架构而出现的。B/S 架构盛行，且随着 SQL Server、MYSQL 等关系型数据库被广泛使用，陆续出现将基于 ASP、PHP、JSP 等网页技术应用于会议室预约系统，但只基于这些技术的系统的开始出现一些不足，如界面显示不太友好，和 HTML 混合在一起可重用性低，不利于维护，因此基于 MVC 架构、SSM 或（SSH）框架的系统都相继出现。MVC 框架强制性地应用程序的输入、处理和输出过程分开，SSH 框架则把多个框架（Struts、Spring、Hibernate）紧密结合在一起，易于扩展，使构建更灵活，界面更友好，且增加了系统信息的安全性。近几年，在人工智能、物联网、云服务等科技发展趋势下，指纹识别、人脸识别的应用越来越广，随着这些技术的发展和应用，智能预约管理成为会议室未来的发展趋势。

## 1.3 课题内容

为了将传统的人工预约管理模式的弊端改善，提高会议室的预约效率，避免资源的冲突，也同时满足一般用户和管理员的需求，本次设计目的在于设计开发一个能够“即时操作、即时见效”，灵活调度，简化预约流程，节省管理员和用户的时间和精力，且实用的在线会议室预约系统。使用所学知识，根据预期的设计，完成系统的开发，开发出的系统应该具有界面美观、操作灵活、使用简便、性能可靠等优点，能够实现普通用户注册登录、查询会议室、预约会议室并修改预约等功能，能够实现管理员登录、管理用户信息、管理会议室等功能。本文就开发系统使用到的相关技术和方法分析基于 B/S 架构、MVC 框架、Java、JSP 等技术的楼宇会议室预约系统。实现会议室预约管理的信息化，改善传统的人工管理纸质记录模式，规范会议室预约的流程，提高管理员对企业会议室预约管理的效率。

## 1.4 论文组织结构

根据研究内容，本文共分为七个章节，详细介绍了楼宇会议室预约系统的设计与实现，具体内容如下：

第一章绪论。介绍了课题背景和国内外现状和趋势，以及本次课题内容。

第二章相关技术与方法介绍。对实现楼宇会议室预约系统所需要的关键技术和结构化系统开发方法进行了介绍。

第三章系统分析。采用结构化系统开发方法分析系统需求，并建立业务流程图、数据流程图及数据字典。

第四章系统设计。根据系统的需求分析，对系统的总体结构、每个功能模块和数据库等进行设计。

第五章系统实现。介绍系统的一些核心功能模块的实现过程，并给出实际效果界面截图和代码截图。

第六章系统测试。对系统功能进行功能性测试，并简要地介绍了其操作流程，最后还以表格记录形式展示了测试结果。

第七章总结。总结本文内容。



## 第二章 相关技术与方法介绍

### 2.1 相关技术介绍

#### 2.1.1 MVC 框架

MVC 的全称是 Model View Controller，是一个框架模式，拥有模型（Model）、视图（View）、控制器（Controller）三个核心部件，这三个部件分别处理各自需要处理的内容，分离业务逻辑和数据显示，体现了便于代码重用的特点。

**Model：**模型层，包含着应用程序的核心数据、逻辑关系以及计算功能，是应用的核心。它封装着程序的业务数据和事务逻辑，为处理操作流程提供了环境，集中呈现了应用程序的状态。

**View：**视图层，是系统与用户交互的界面，它许可用户在页面上做编辑，然后把系统的处理结果以一定形式反馈给用户，但是它只能算是模型的外在形式，不进行任何实际的业务操作。每个视图都含有一个更新操作，当使用此操作时，视图截取来自模型的数据值，然后运用它来更新显示。

**Controller：**控制器，负责控制视图和模型的交互过程。当控制器获取到用户请求后，就使用模型调用去实现这个请求，然后再根据实现响应的视图，给用户展现模型进行更新操作后产出的数据。

图 2-1 展示了 MVC 框架图。

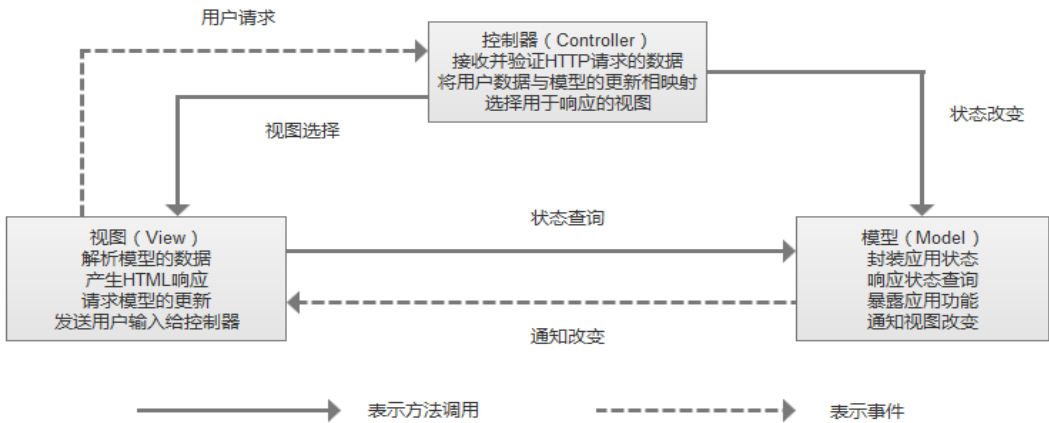


图 2-1 MVC 组件关系图

## 2.1.2 B/S 架构

B/S 架构全称为 **Browser/Server** 架构，即浏览器/服务器架构。实际上是一种典型三层体系结构的应用，是 **WEB** 兴起流行后的一种网络结构模式。第一层客户端浏览器作为用户与整个系统的接口，它将代码转化成有图有文且具备交互功能的网页，允许用户在网页表格里输入信息提交给后台作出处理请求。第二层 **Web** 服务器端主要负责处理数据，或者实现各种业务功能。它会启动对应的进程来响应用户的请求，然后动态地生成一串嵌入了处理结果的代码，返回给客户端浏览器。第三层数据库服务器层主要负责协调不同服务器发出来的请求和管理系统数据库。三层 B/S 体系架构图如图 2-2 所示。



图 2-2 三层 B/S 体系架构图

## 2.1.3 JSP

JSP（**JAVA SERVER PAGES**），即 **Java** 服务器页面，是 **Sun** 公司所发明的一种动态网页技术标准。JSP 的脚本语言是 **Java**，整个服务器端服务于 **HTTP** 应用程序都由这一网页提供接口。其主要包括静态和动态两个部分，静态部分的编写和用普通的 **HTML** 语言编写网页一样，动态部分则在“<% %>”里，它的内容的编写基于 **Java** 语言，遵从 **Java** 的语法。JSP 部署在网络服务器上，引入“JSP 动作”的 **XML** 标签，用来内建功能，可及时响应客户端发送的请求，按照请求内容动态地生成 **HTML** 或者其他格式的 **Web** 网页，并返回给请求者。JSP 文件后缀名为\*.jsp，JSP 技术以 **Servlet** 为基础，其本身会被预先编译成 **Servlet**，之后就不需要再重新编译则可直接运行了。JSP 开发的 **WEB** 应用可跨平台使用，同时支持 **Linux** 系统和 **Windows** 系统。

## 2.2 系统开发方法概述

结构化系统开发方法是指采用系统的思想、采用系统工程的一种开发方法，遵循用户至上原则，让分析人员能够以逻辑方式了解系统及其主要活动，是一种从高级概述到较低级别详细信息的方法。其是应用最广泛的开发方法，实质是着眼于数据流，采用自顶向下分析与设计，然后自底向上逐步实施建立信息系统的处理流程，把数据流图和数据字典作为主要工具，建立系统的逻辑模型。用该方法开发一个系统，可将整个开发过程分为几个阶段，然后逐步实现，前一阶段为后一阶段的工作提供基础，而每个阶段又分为详细的工作步骤和顺序操作。

本系统具体的结构化分析步骤如下：

- （1）通过对目前国内外现状的调查，以用户的需求为线索，获得当前系统具体的物理模型；
- （2）根据物理模型推导出当前系统的逻辑模型；
- （3）设计新的逻辑系统，生成自顶向下的数据流程图和数据字典；
- （4）对系统进行功能模块结构设计、数据库设计；
- （5）详细设计组合实现一个合理的、简便的、符合需求的楼宇会议室预约系统。

## 2.3 本章小结

本章主要介绍了系统开发中常用的 MVC 框架，B/S 架构，JSP 等技术，并对结构化系统开发方法作了简要概述，为后续项目的设计与实现奠定了基础。

## 第三章 系统分析

### 3.1 系统需求概述

楼宇会议室的预约，离不开企业高效、规范、便捷的管理。然而，随着网络技术的发展，公司业务的多样化和规模逐渐扩大，传统的人工管理记录方式的弊端越发明显，存在诸多问题。首先，尽管企业的信息化工作已经开展多年，但企业内部会议室的预约管理却往往被忽略，甚至有的企业并没有专门的会议室预约系统，预约通常需要经过预约咨询、填送报表、签字盖章等繁琐的过程，不仅使得工作效率低下，也使得管理员工作量极大增加。其次，传统的会议室预约一般采用人工管理方式，无法严格审查预约时间是否会产生冲突的现象，多个部门在同一时段申请会议室的情况也有可能出现。最后，采用这种陈旧的预约管理方式，过程繁琐，用户无法及时准确的确定会议室的使用情况，发生冲突后查找已被占用和未被占用的会议室比较困难，无法进行及时的协调和修改，存在不够灵活和不够高效等弊端，很大程度上影响了企业的工作效率。因此，通过引入楼宇会议室预约系统，可以很好的实现会议室预约的信息化，规范预约流程，提高企业部门工作的高效性和便捷性。此楼宇会议室预约系统主要实现的功能有：用户注册、用户登录、个人资料管理、系统用户管理、会议室查询、在线留言、会议室预约、会议室管理、预约更改、站内新闻管理、会议室预约管理、系统管理。

### 3.2 业务流程调查与分析

业务流程图是说明系统内部各部门与人员之间的业务关系和作业顺序以及管理信息流向的流程图，属于一种物理模型，是描述部门内部业务处理活动内容的工作图。

本系统的业务流程为：首先进行用户选择，用户为管理员就登录到后台进行相关信息的管理，用户为普通用户就通过注册，若注册成功就可以登录进系统前台进行功能操作，完成预约等。图 3-1 展示了本系统的业务流程图。

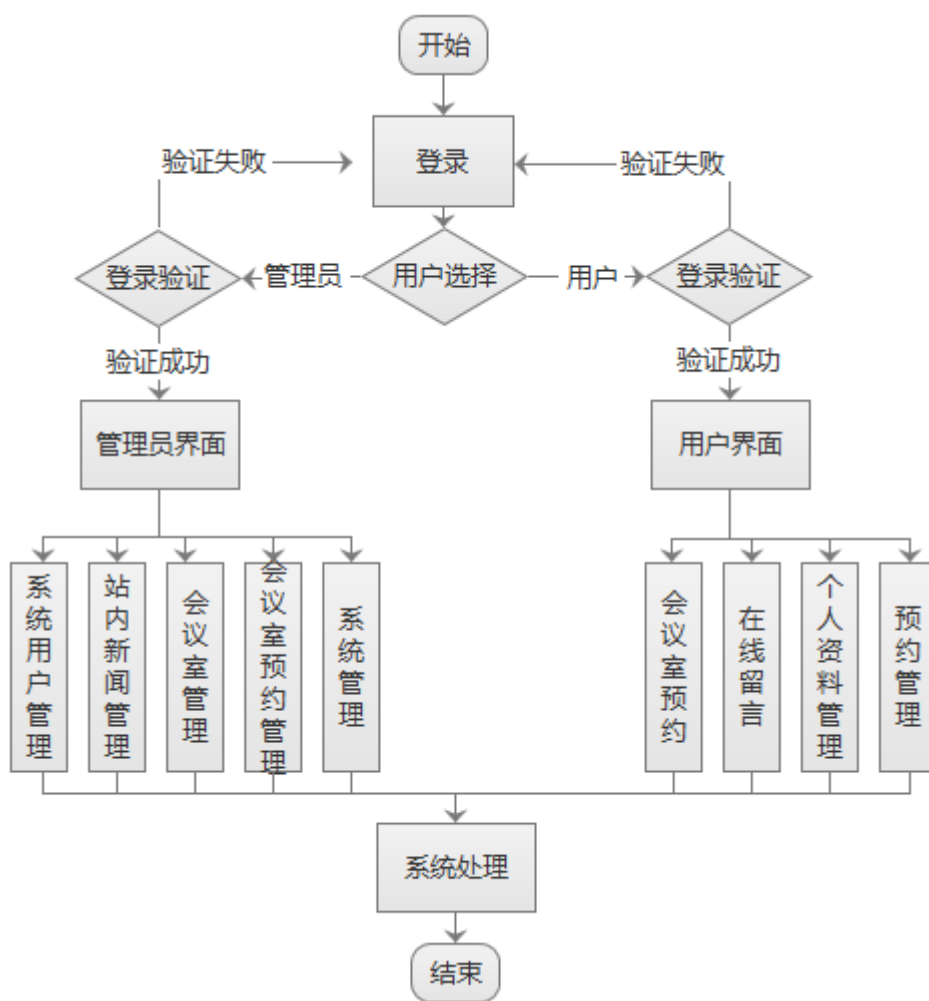


图 3-1 系统业务流程图

### 3.3 系统逻辑模型

#### 3.3.1 数据流程分析

数据流程图是用来说明信息系统数据流程的工具，由复杂的数据独立抽象而成，图中无具体的物理元素，它可用一些符号概括性地反映出信息在系统中的流动、处理和存储情况，具有概括性和抽象性。

顶层数据流程属于流程中最抽象的一层，它简要描述了系统角色即管理员和用户与系统之间的大致交互流程。

图 3-2 展示了本系统的顶层数据流程图。

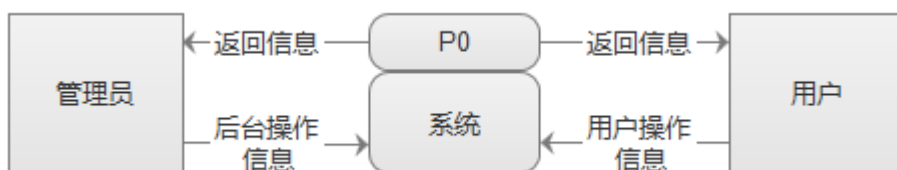


图 3-2 系统数据流程图（顶层）

一层数据流程细化了顶层数据流程，将整体系统分为用户操作部分和管理员操作部分展开说明，并给出了各角色用户各在两部分的信息操作流程。

系统一层数据流程图如图 3-3 所示。

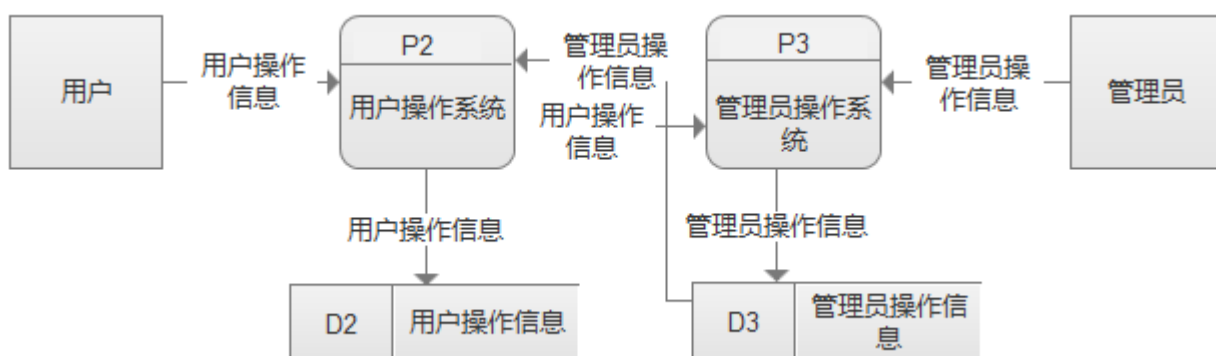


图 3-3 系统数据流程图（一层）

二层数据流程对一层数据流程中用户操作部分和管理员操作部分进行了细化，即用户操作细分为留言、会议室预约、查新新闻、查看公告等，管理员操作细分为留言管理、会议室管理、会议室预约管理等。

图 3-4 展示了本系统的二层数据流程图。

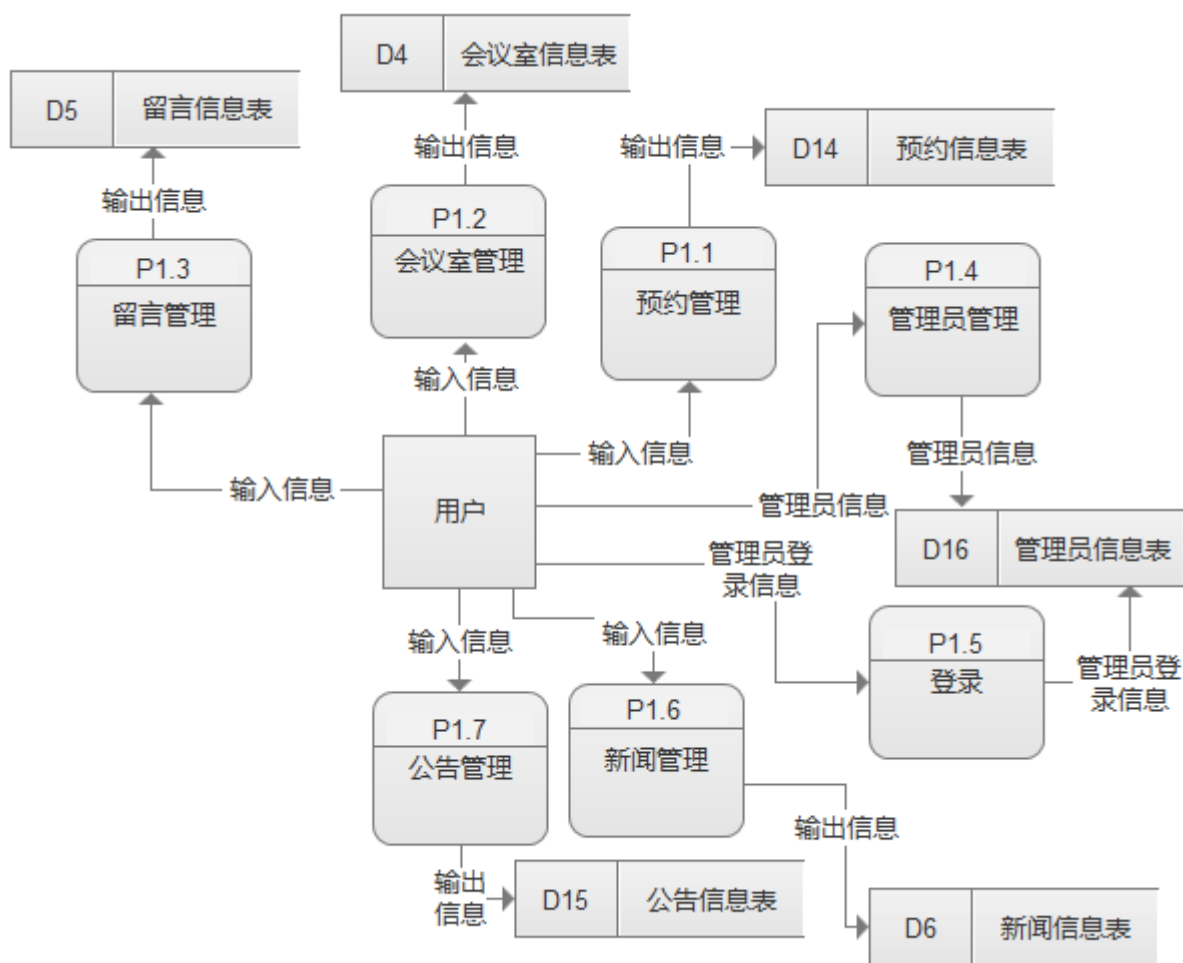


图 3-4 系统数据流程图（二层）

### 3.3.2 数据字典

数据字典主要是对数据的数据项、数据结构、数据流、数据存储和数据处理等过程作出说明，能详细阐述一个数据流程图。其是人们了解系统的一个重要工具，它将数据中的所有元素的定义都集合起来，成为后续数据库设计的一个重要参考。

#### （1）数据流

表 3-1 用户注册信息表

数据流名称：用户注册信息

来源：用户注册

去向：用户信息表

组成：用户名+密码+姓名+性别+出生年月+QQ+邮箱+电话+身份证+头像+地址+备注+是否审核

表 3-2 管理员信息表

数据流名称：管理员登录信息

来源：管理员登录

去向：管理员信息表

组成：用户名+密码+权限

## (2) 数据存储

表 3-3 会议室信息表

数据存储名称：会议室信息表

来源：会议室管理

去向：会议室统计表

组成：会议室编号+会议室名称+会议室类型+会议室面积+图片+备注

存储方式：顺序



表 3-4 会议室预约信息表

数据存储名称：会议室预约信息表

来源：会议室预约

去向：会议室预约统计表

组成：会议室编号+会议室名称+会议室类型+会议室面积+预约时间+预约时段+预约用户+备注+是否审核

存储方式：顺序

表 3-5 留言信息表

数据存储名称：留言信息表

来源：在线留言

去向：留言统计表

组成：用户名+性别+QQ+邮箱+电话+内容+回复

存储方式：顺序

表 3-6 新闻信息表

数据存储名称：新闻信息表

来源：新闻通知

去向：新闻统计表

组成：标题+类别+内容+添加人+首页图片+点击率

存储方式：顺序

### 3.4 本章小结

本章主要对楼宇会议室预约进行了用户需求分析和概述，指出设计原则，然后作出了系统的业务流程图、和三层数据流程图以及规划了数据字典。

## 第四章 系统设计

### 4.1 系统设计概述

考虑到本项目中系统流程的多样性和复杂性，在设计系统时提出了应遵循的设计原则主要有：（1）模块化，为了系统后期的维护更加便捷，应将系统功能划分为相对独立的功能模块，不但能提高开发效率，还能有效地降低耦合程度。（2）权限控制，考虑到使用本系统有两种角色，即普通用户和管理员，因此应该引入权限控制思想，为各角色分配不同的功能权限，以限制用户操作权限，保障了系统应用和数据的安全性。（3）跨平台性，即系统不需要依赖于特定的硬件平台，该系统可以被使用于任何操作系统。

本系统遵循 MVC 框架，采用分层设计模式进行设计，把整个系统实现分成表示层、业务层和持久层。表示层是用户和系统进行互动的层，一般是实现前端各页面间的跳转，并使用业务层提供的服务来满足用户的需求。在这一层只需编写 JSP、ACTION 代码，并利用 CSS、HTML、JavaScript 等技术让前端页面得到美化。业务层由一个个 Service 组成，每个 Service 层间是相互独立的，它们是作为程序的特定模块来实现应用功能的，处理时主要是采用 DAO 层方法来操作后台数据库，然后将数据库中的数据传到前台给用户作出显示。持久层负责屏蔽底层数据库的连接，提供一个简洁、统一和面向对象的数据访问接口给业务层，让各层间以对象方式传递参数值，极大简化了代码的可读性，减少编写重复代码的操作。

图 4-1 展示了系统整体架构的构件图。

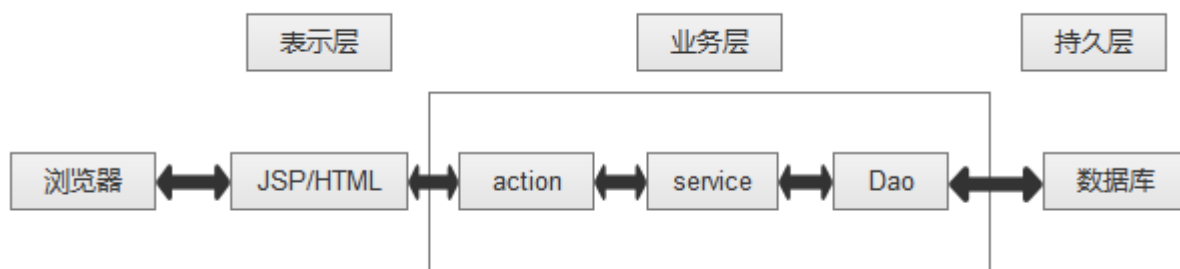


图 4-1 整体架构

## 4.2 系统总体设计

本系统有普通用户和管理员两种角色，系统操作分为前台和后台两个部分。在前台，普通用户先填写基本信息注册，等待管理员审核，通过了才能输入账号密码登录到系统前台，然后查询会议室等信息，在主页面的菜单栏中可以直接选择在线留言、预约会议室等，还能在首页点击个人后台进入个人资料管理、预约管理等页面进行操作，可在前台的搜索框内输入标题或者类别搜索信息。

在后台，管理员在首页点击管理员的按钮，进入管理员登录页面，输入账号和密码，然后等待数据库身份验证，验证成功后进入后台主页，实现系统用户管理、会议室管理、会议室预约管理等功能。管理员还可以在会议室管理、会议室预约管理等页面通过关键字查找信息，并将查找出的信息列表导出 excel 中。

图 4-2 展示了系统的功能结构。

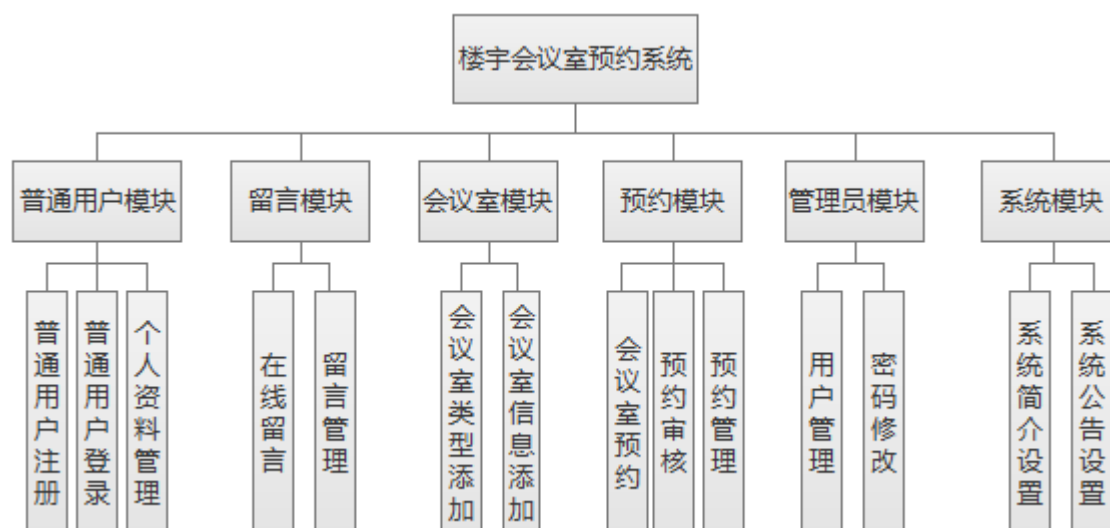


图 4-2 系统功能结构图

## 4.3 功能模块详细设计

### 4.3.1 普通用户模块

这个模块分为普通用户注册、普通用户登录、个人资料管理等部分，普通用户先在注册页面填写相应的信息，然后等待管理员审核，若审核通过则表示注册成功，可登录系统进行操作，在后期若个人信息有所更改，还可进入个人资料管理修改个

人信息。

普通用户注册流程图如图 4-3 所示。

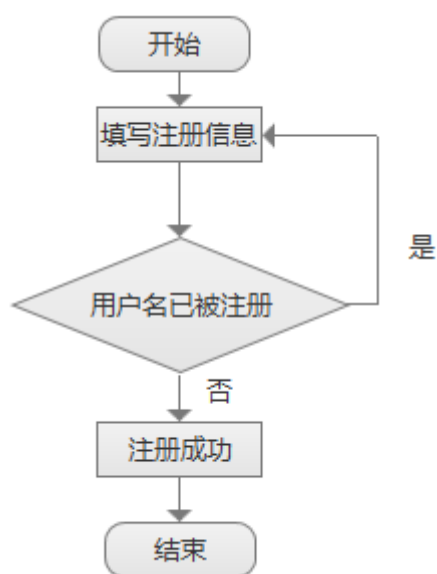


图 4-3 普通用户注册流程图

### 4.3.2 留言模块

该模块分为在线留言和留言管理两部分，在线留言由注册成功的用户登录系统后实现，留言管理由管理员登录系统后进入系统管理页面，查看已发布的留言内容然后可选择回复或删除等。

在线留言流程图如图 4-4 所示。

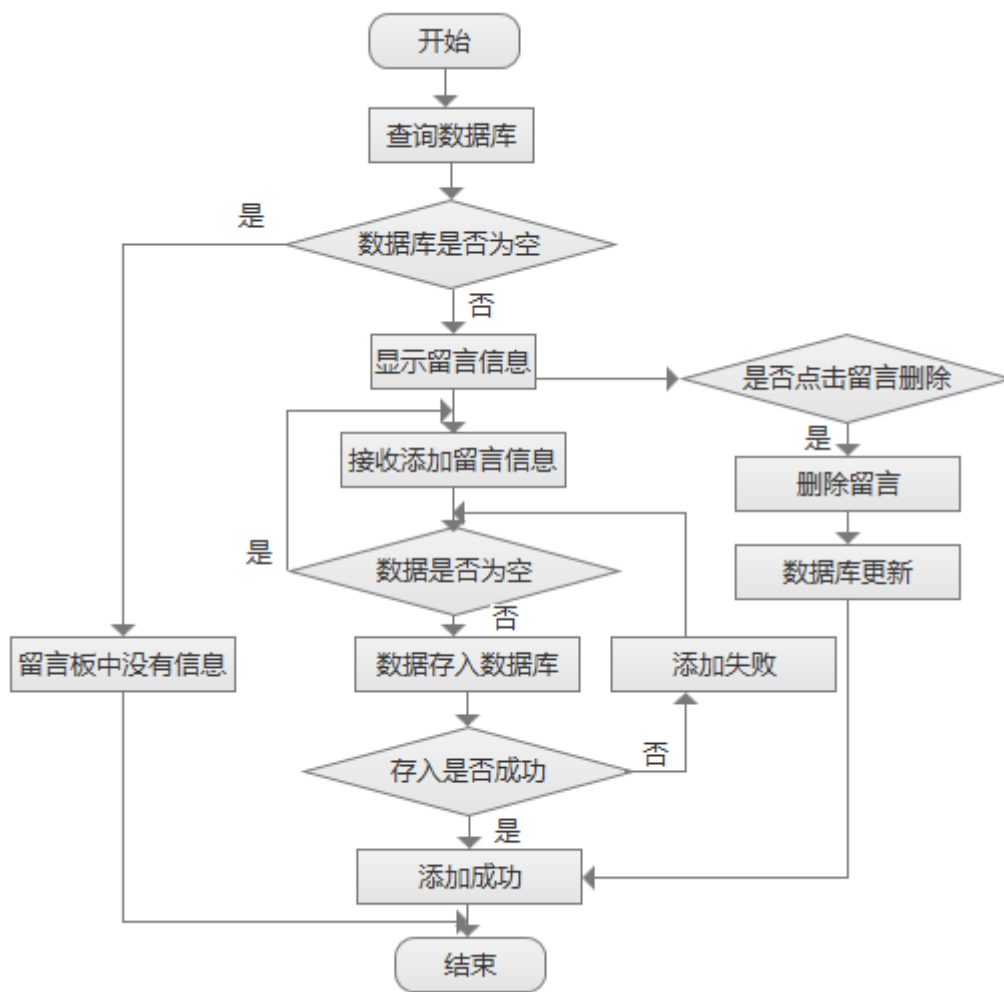


图 4-4 在线留言流程图

### 4.3.3 会议室模块

会议室模块分为会议室类型管理、会议室信息管理两部分，会议室类型主要分为 A、B、C 类，会议室信息主要是编号、名称、类型、面积、图片等内容。

会议室信息添加流程图如图 4-5 所示。

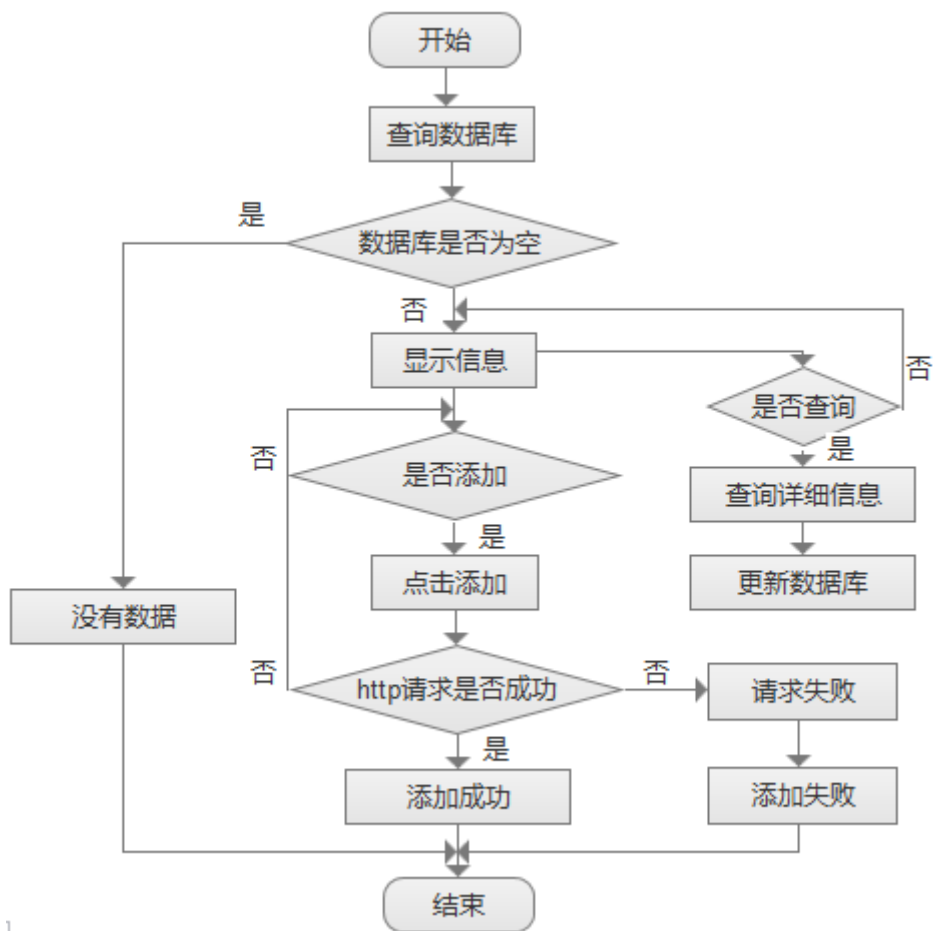


图 4-5 会议室信息添加流程图

#### 4.3.4 预约模块

预约模块分为预约申请、预约审核、预约管理等部分，会议室预约和预约管理都是由普通用户实现，普通用户点击会议室信息详情页，点击预约按钮后进入预约页面，选择预约时间和时段提交，然后由管理员登录到预约审核页面，查看普通用户的预约并进行审核，若审核通过则表示普通用户已成功实现预约。

图 4-6 展示了会议室预约的流程。

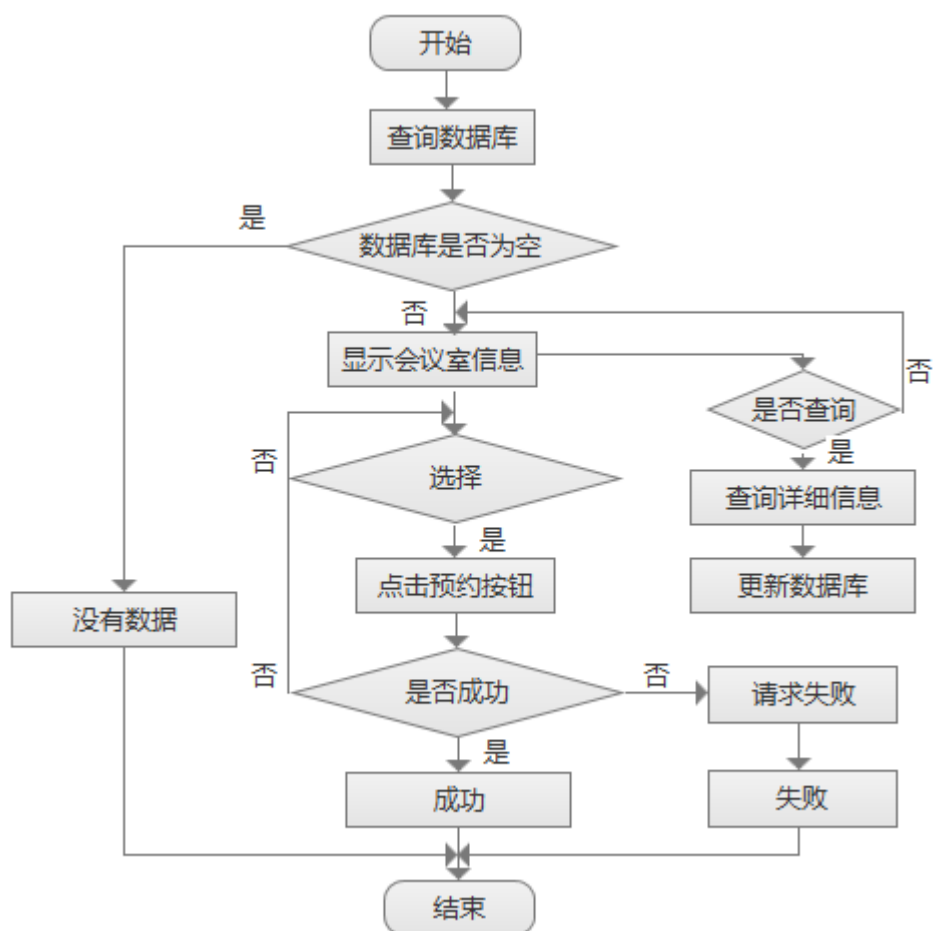


图 4-6 会议室预约流程图

### 4.3.5 管理员模块

这个模块有用户管理和密码修改，用户管理部分又分为普通用户管理和管理员管理。在普通用户管理部分，管理员可进入已有用户注册列表页面查看或者根据关键信息搜索，可对已注册用户的信息进行修改，也可直接删除注册用户，以及对用户的注册申请进行审核等。在管理员管理部分，主要负责添加新管理员，或将已存在的管理员进行删除等操作。密码修改部分是用新密码替换原密码，管理员在输入正确的原密码后，还要输入两次新密码，第二次新密码的输入是对新密码的再次确认。

密码修改流程图如图 4-7 所示。

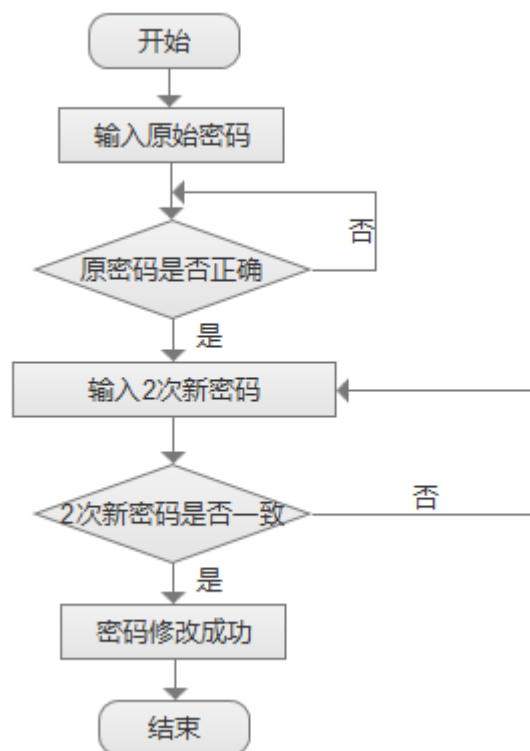


图 4-7 密码修改流程图

### 4.3.6 系统模块

系统模块分为系统简介设置和系统公告设置，目的是让使用者能够更快的了解系统以及接收到系统的一些日常通知。

## 4.4 数据库设计

### 4.4.1 概念设计

数据库的前期设计中，概念结构的设计是其最关键的一步。只有将需求分析抽象成信息世界的结构，才能更合理的使用 DBMS。此设计具有体现事物之间关联关系和便于理解更改数据库两个优势。

E-R 模型是描述概念模型最有力的工具。E-R 图也称实体-联系图，是用来描述现实世界的概念模型，使用它能够清晰的指导每个实体之间的联系，实体、属性和联系是它最关键的三个元素。

图 4-8 展示了系统的总体 E-R 图。



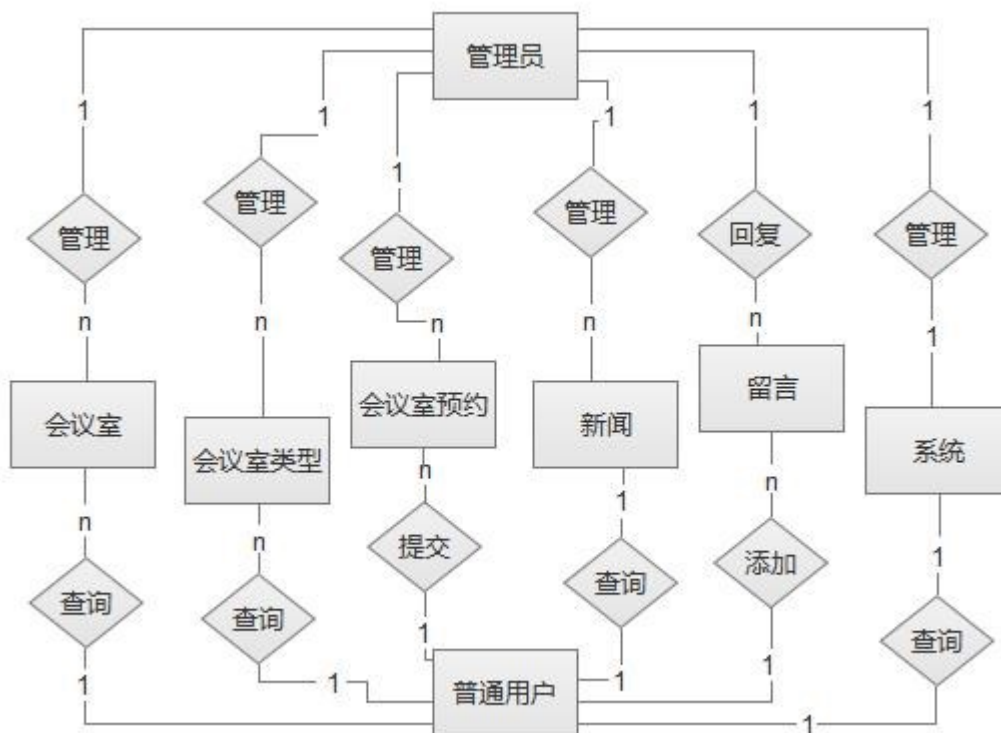


图 4-8 系统总体 E-R 图

## 4.4.2 逻辑设计

数据库由概念模型转为逻辑模型的过程也可以称为逻辑设计，即就是将实体图转换成以关系模式或层次模式呈现出来的形式。本系统的数据库逻辑结构如下。

### (1) 管理员

序号（主键），用户名，密码，权限。

### (2) 普通用户

序号（主键），用户名，密码，姓名，性别，出生年月，QQ，邮箱，电话，身份证，头像，地址，备注，是否审核。

### (3) 会议室类型

序号（主键），类型，添加时间。

### (4) 会议室信息

序号（主键），会议室编号，会议室名称，会议室类型，会议室面积，图片，备注，添加时间。

### (5) 会议室预约

序号（主键），会议室编号，会议室名称，会议室类型，会议室面积，预约时间，预约时段，预约用户，备注，是否审核，添加时间。

（6）系统信息

序号（主键），类别，内容。

（7）留言信息

序号（主键），用户名，性别，QQ，邮箱，电话，内容，添加时间，回复。

（8）新闻通知

序号（主键），标题，类别，内容，添加人，首页图片，点击率。

### 4.4.3 物理结构设计

数据库的物理结构设计，可以理解为在数据库中创建表，然后设置表的属性和大小，建立表与表间的关联关系等操作。根据第 4.4.1 节设计的 E-R 图和第 4.4.2 节设计的逻辑模型，本系统所用到的数据表主要有：管理员信息表，普通用户信息表，会议室类型信息表，会议室信息表，会议室预约信息表，系统信息表，留言信息表，新闻通知信息表等 8 张表，下面将对各表进行介绍。

（1）管理员信息表

管理员信息表的结构如表 4-1 所示。

表4-1 管理员信息表

字段名	数据类型	说明
ID	int(10)	自增主键
username	varchar(255)	管理员用户名
pwd	varchar(255)	管理员登录密码
qx	varchar(255)	管理员权限
addtime	datetime(23)	管理员添加时间

（2）普通用户信息表

普通用户信息表的结构如下表4-2所示。

表4-2 普通用户信息表

字段名	数据类型	说明
ID	int(10)	自增主键
yonghuming	varchar(255)	用户名

mima	varchar(255)	密码
xingming	varchar(255)	姓名
xingbie	varchar(255)	性别
chushengnianyue	varchar(255)	出生年月
QQ	varchar(255)	QQ
youxiang	varchar(255)	邮箱
dianhua	varchar(255)	电话
shengfenzheng	varchar(255)	身份证
touxian	varchar(255)	头像
dizhi	varchar(255)	地址
beizhu	varchar(255)	备注
addtime	datetime(23)	添加时间
shifoushenhe	varchar(255)	是否审核

### (3) 会议室类型信息表

会议室类型信息表的结构如下表4-3所示。

表4-3 会议室类型信息表

字段名	数据类型	说明
ID	int(10)	自增主键
leixing	varchar(255)	会议室类型
addtime	datetime(23)	添加时间

### (4) 会议室信息表

会议室信息表的结构如下表4-4所示。

表4-4 会议室信息表

字段名	数据类型	说明
ID	int(10)	自增主键
changdibianhao	varchar(255)	会议室编号
changdimingcheng	varchar(255)	会议室名称
changdileixing	varchar(255)	会议室类型
changdimianji	varchar(255)	会议室面积
tupian	varchar(255)	会议室图片
beizhu	varchar(255)	备注
addtime	datetime(23)	添加时间

(5) 会议室预约信息表

会议室预约信息表的结构如下表4-5所示。

表4-5 会议室预约信息表

字段名	数据类型	说明
ID	int(10)	自增主键
changdibianhao	varchar(255)	会议室编号
changdimingcheng	varchar(255)	会议室名称
changdileixing	varchar(255)	会议室类型
changdimianji	varchar(255)	会议室面积
yuyueshijian	varchar(255)	预约时段
yuyueren	varchar(255)	预约人
beizhu	varchar(255)	备注
shifoushenhe	varchar(255)	是否审核
addtime	datetime(23)	添加时间

(6) 系统信息表

系统信息表的结构如下表4-6所示。

表4-6 系统信息表

字段名	数据类型	说明
ID	int(10)	自增主键
leibie	varchar(255)	信息类别
content	varchar(255)	内容

(7) 留言信息表

留言信息表的结构如下表4-7所示。

表4-7 留言信息表

字段名	数据类型	说明
ID	int(10)	自增主键
mingcheng	varchar(255)	留言昵称
xingbie	varchar(255)	性别
QQ	varchar(255)	QQ
youxiang	varchar(255)	邮箱
dainhua	varchar(255)	电话
neirong	varchar(255)	内容
addtime	datetime(23)	添加时间

huifuncirong	varchar(255)	回复内容
--------------	--------------	------

#### (8) 新闻通知信息表

新闻通知信息表的结构如下表4-8所示。

表4-8 新闻通知信息表

字段名	数据类型	说明
ID	int(10)	自增主键
baioti	varchar(255)	新闻标题
leibie	varchar(255)	新闻类别
neirong	varchar(255)	新闻内容
tianjiaren	varchar(255)	新闻添加人
shouyetupian	varchar(255)	新闻首页图片

## 4.5 本章小结

本章前部分先对系统的设计原则进行确定，然后对系统的结构进行了分析设计，并给出了相应操作的流程图，最后对数据库表进行了详细设计。

## 第五章 系统实现

### 5.1 项目系统层次设计

楼宇会议室预约系统项目层次结构如图 5-1 所示。

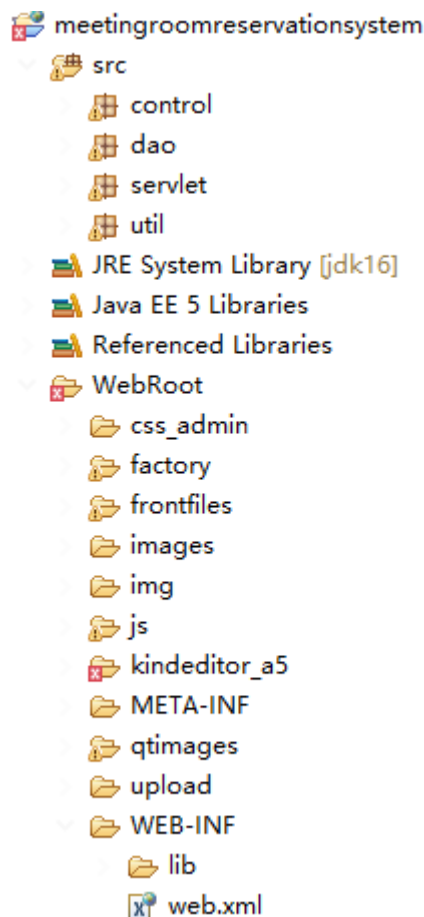


图 5-1 系统项目层次图

**control 层。**集合各种 action、业务模块流程，控制着用户的输入。

**dao 层。**主要负责数据库交互的工作，与数据库进行联系的一些任务都封装在这一层，其拥有最多的处理任务，主要处理应用程序数据逻辑部分。

**servlet 层。**主要负责获取用户的输入信息，然后调用模型和视图去完成用户需求，其本身不输出任何东西或者做任何处理，而是接收请求然后决定要调用的构建模型去处理请求，并合理确定好所使用的视图来将数据显示出来。主要用于页面的跳转。

util。主要负责存放系统使用的一些工具类。

## 5.2 各层的具体实现

control 层主要负责控制着用户的输入还有处理一些业务模块流程，如对于登录的普通用户，登录成功的前提是已成为注册用户，若不是已注册用户，则无法登录。实现代码如下图 5-2、图 5-3 所示。

```
if(ac.equals("login"))
{
    String username = request.getParameter("username");
    String password = request.getParameter("pwd1");
    String utype = request.getParameter("cx");
    String pagerandom =
request.getParameter("pagerandom")==null?"":request.getParameter("pagerandom");
    String random = (String)request.getSession().getAttribute("random");
    if(!pagerandom.equals(random)&&request.getParameter("a")!=null)
    {
        request.setAttribute("random", "");
        go("/index.jsp", request, response);
    }
    else{
        String sql1="";
```

图 5-2 control 层实现代码（1）

```
        if (utype.equals("注册用户")){sql1 = "select * from yonghuzhuce
where yonghuming='"+username+"' and mima='"+password+"' and issh='是'";}
        List<HashMap> userlist1 = dao.select(sql1);
        if(userlist1.size()==1)
        {
            request.getSession().setAttribute("username", username);
            request.getSession().setAttribute("cx", utype);
            go("/index.jsp", request, response);
        } else {
            request.setAttribute("error", "");
            go("/index.jsp", request, response);
        }
    }
}
```

图 5-3 control 层实现代码（2）

dao 层主要是实现与数据库的交互，从数据库表中提取数据或者将数据存入表

中都在此处实现。连接数据库的实现代码如下图 5-4 所示。

```
public Connection getConn()
{
    try
    {
        if(conn==null||conn.isClosed()){

            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection
("jdbc:mysql://localhost:3306/jspmhycdyyxtglhsg31868994mysql","root","root");
        }
    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
    return conn;
}
```

图 5-4 数据库连接实现代码

servlet 层主要负责页面的跳转，实现代码如下图 5-5 所示。

```
public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    request.setCharacterEncoding("utf-8");
    response.setCharacterEncoding("utf-8");
    String website = request.getParameter("website");
    try {
        QRCodeUtil.genQR(website, response.getOutputStream());
    } catch (WriterException e) {
        e.printStackTrace();
    }
}

public void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
```

图 5-5 servlet 层实现代码



## 5.3 系统核心功能模块实现

### 5.3.1 前台首页的实现

网站的主页面先显示出所有功能的水平菜单栏供用户选择，中间是最近的新闻，下方是滚动的会议室图片条。用户在主界面可以看到注册登录处，没有注册的用户则点击用户注册按钮进入到注册页面填写信息，已注册成功的用户则在页面左侧填写用户名和密码，并输入正确的验证码提交登录。在页面的下方是热门会议室推荐的滚动图片显示。在会议室较多的情况下，为了方便用户对某一心意相符的特定会议室信息的查看，在页面的左下方提供了查询的输入栏，通过用户输入关键字来进行会议室信息的特定查询。主界面的显示效果如图 5-6、图 5-7 所示。



图 5-6 系统主页面（1）



图 5-7 系统主页面（2）

热门会议室的图片滚动显示可以让用户得到更好的视觉体验。实现代码如下图 5-8 所示。

```
<%
for(HashMap m:new CommDAO().select("select * from changdixinxi order by id
desc",1,8)){
%>
<td align=center width="190"><a
href="changdixinxidetail.jsp?id=%m.get("id")%"><br>
<br><%m.get("changdimingcheng") %></a></td>
%> %>

. . . . .
<script type="text/javascript">
var speed=50
marqueePic2.innerHTML=marqueePic1.innerHTML
function Marquee(){
if(demo.scrollLeft>=marqueePic1.scrollWidth){
demo.scrollLeft=0
} else {
demo.scrollLeft++
}
}
var MyMar=setInterval(Marquee,speed)
demo.onmouseover=function() {clearInterval(MyMar)}
demo.onmouseout=function() {MyMar=setInterval(Marquee,speed)}
</script>
```

图 5-8 热门会议室实现代码

### 5.3.2 普通用户注册功能的实现

在进入主页面之后，为普通用户提供了两种选项，如果目前的这个用户还没有账号，那么该用户就需要进行注册，注册成功以后才可继续进行登录验证操作。用户进入到注册页面，并根据提示填写用户名、密码、姓名等信息，在用户名未被注册使用过且其他信息填写符合要求下才能提交成功。然后由业务层将信息保存到普通用户注册表中，注册成功后系统会给出提示信息。下图 5-9 展示了普通用户的注册页面。

用户注册

用户名:	<input type="text" value="lili98"/>		
密码:	<input type="password" value="....."/>	确认密码:	<input type="password" value="....."/>
姓名:	<input type="text" value="李曼"/>		
性别:	<input type="text" value="女"/> ▼		
出生年月:	<input type="text" value="1998-02-05"/>		
QQ:	<input type="text" value="1396965345"/>		
邮箱:	<input type="text" value="1396965345@qq.com"/>		
手机:	<input type="text" value="13649407177"/>		
身份证:	<input type="text" value="452126199802051528"/>		
头像:	<input type="text" value="upload/1589013043926.jpg"/>	<input type="button" value="上传"/>	
地址:	<input type="text" value="江苏省苏州市姑苏区东环路50号苏州大学"/>		
备注:	<div>无。</div>		
<div><input type="button" value="提交"/> <input type="button" value="重置"/></div>			

图 5-9 普通用户注册页面

普通用户注册成功提示如图 5-10 所示。



图 5-10 普通用户注册成功提示页面

普通用户在输入完个人信息后，点击提交完成注册操作，后台在接收到数据信息后，会自动将字段信息映射到对应的对象属性中，减少后代代码的编写同时完成数的传输，实现代码如下图 5-11、图 5-12、图 5-13、图 5-14 所示。

```
<%  
HashMap ext = new HashMap();  
ext.put("issh","否");  
new  
CommDAO().insert(request,response,"yonghuzhuce",ext,true,false,"index.jsp");  
%>  
  
<table width="98%" border="1" align="center" cellpadding="3" cellspacing="1"  
bordercolor="#58AC22" style="border-collapse:collapse"  
<tr>  
  <td width="15%">用户名: </td>  
  <td colspan="2"><input name='yonghuming' type='text' id='yonghuming'  
value="" onblur='checkform()' /><label id='clabelyonghuming' />  
&nbsp;*</td>  
</tr>  
<tr>  
  <td>密码: </td>  
  <td width="27%"><input name='mima' type='password' id='mima' value=""  
onblur='checkform()' /><label id='clabelmima' />  
&nbsp;*</td>  
  <td width="58%">确认密码:  
    <input name='mima2' type='password' id='mima2' value=""  
onblur='checkform()' /> <label id='clabelmima2' /> </td>  
</tr>
```

图 5-11 用户注册实现代码（1）



```

</tr>
<td>姓名: </td>
<td colspan="2"><input name='xingming' type='text' id='xingming' value=""
onblur='checkform()' /><label id='clabelxingming' />
&nbsp;*</td>
</tr>
<tr>
<td>性别: </td>
<td colspan="2"><select name='xingbie' id='xingbie' style=" height:19px;
border:solid 1px #000000; color:#666666">
<option value="男">男</option>
<option value="女">女</option>
</select></td>
</tr>
<tr>
<td>出生年月: </td>
<td colspan="2"><input name='chushengnianyue' type='text'
id='chushengnianyue' value="" readonly='readonly'
onClick="WdatePicker({'dateFmt':'yyyy-MM-dd'})" /></td>

```

图 5-12 用户注册实现代码（2）

```

</tr>
<tr>
<td>QQ: </td>
<td colspan="2"><input name='QQ' type='text' id='QQ' value=""
onblur='checkform()' /> <label id='clabelQQ' /></td>
</tr>
<tr>
<td>邮箱: </td>
<td colspan="2"><input name='youxiang' type='text' id='youxiang' value=""
onblur='checkform()' /> <label id='clabelyouxiang' /></td>
</tr>
<tr>
<td>手机: </td>
<td colspan="2"><input name='dianhua' type='text' id='dianhua' value=""
onblur='checkform()' /> <label id='clabeldianhua' /></td>
</tr>
<tr>
<td>身份证: </td>
<td colspan="2"><input name='shenfenzheng' type='text' id='shenfenzheng'
value="" onblur='checkform()' /> <label id='clabelshenfenzheng' /> </td>
</tr>

```

图 5-13 用户注册实现代码（3）

```

<tr>
  <td>头像: </td>
  <td colspan="2"><input name='touxiang' type='text' id='touxiang' size='50'
value="" />
  &nbsp;
  <input name="button" type='button' onClick="up('touxiang')" value='上传'
  /></td>
</tr>
<tr>
  <td>地址: </td>
  <td colspan="2"><input name='dizhi' type='text' id='dizhi' value="" size='50'
  /></td>
</tr>
<tr>
  <td>备注: </td>
  <td colspan="2"><textarea name='beizhu' cols='50' rows='5'
id='beizhu'></textarea></td>
</tr>
<tr>
  <td>&nbsp;</td>
  <td colspan="2"><input type="submit" name="Submit5" value="提交"
onClick="return check();"/>

```

图 5-14 用户注册实现代码（4）

### 5.3.3 会议室预约功能的实现

用户在页面查找会议室或直接点击预约会议室按钮，进入会议室信息浏览页，选择自己想要预约的会议室然后点击详细进入会议室信息详情页，普通用户在详情页可以点击预约按钮填写预约申请，预约提交成功后，管理员还需要查看普通用户的预约信息，然后审核。

下图 5-15 展示了会议室信息界面。

会议室信息						
搜索：会议室编号： <input type="text"/> 会议室名称： <input type="text"/> <input type="button" value="查找"/> <input type="button" value="切换视图"/>						
序号	会议室编号	会议室名称	会议室类型	会议室面积	图片	详细
1	0003	会议室3	C类	2225		<a href="#">详细</a>
2	0002	会议室2	C类	1000		<a href="#">详细</a>
3	0001	会议室1	B类	2225		<a href="#">详细</a>
共3条 第1/1页 <a href="#">首页</a> <a href="#">上页</a> <a href="#">下页</a> <a href="#">末页</a> <input type="text" value="1"/> <input type="button" value="v"/>						

图 5-15 会议室信息页

下图 5-16 展示了会议室预约界面。

会议室预约	
会议室编号：	<input type="text" value="0001"/>
会议室名称：	<input type="text" value="会议室1"/>
会议室类型：	<input type="text" value="B类"/>
会议室面积：	<input type="text" value="2225"/>
预约时间：	<input type="text" value="2020-05-13"/>
预约时段：	<input type="text" value="13:00-14:30"/> <input type="button" value="v"/>
预约人：	<input type="text" value="lili98"/>
备注：	<input type="text" value="无"/> <input type="button" value="x"/>
	<input type="button" value="提交"/> <input type="button" value="重置"/>

图 5-16 会议室预约界面

当预约成功后，会弹出一个提示窗口显示操作成功，如图 5-17 所示。

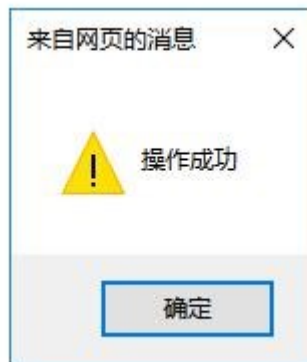


图 5-17 预约成功提示

获取会议室 id 的 getmaps 方法实现代码如下图 5-18、图 5-19 所示。

```
public HashMap getmaps(String nzd,String zdz,String table)
{
    List<HashMap> list = new ArrayList();
    try {
        Statement st = conn.createStatement();
        ResultSet rs = st.executeQuery("select * from "+table+" where
"+nzd+"='"+zdz+"'");
        ResultSetMetaData rsm = rs.getMetaData();
        while(rs.next())
        {
            HashMap map = new HashMap();
            int i = rsm.getColumnCount();
            for(int j=1;j<=i;j++)
            {
                if(!rsm.getColumnName(j).equals("ID"))
                {
                    String str = rs.getString(j)==null?"":rs.getString(j);
                    if(str.equals("null"))str = "";
                    map.put(rsm.getColumnName(j), str);
                }
                else
                    map.put("id", rs.getString(j));
            }
            list.add(map);
        }
    }
}
```

图 5-18 getmaps 方法实现代码（1）



```

        rs.close();
        st.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return list.get(0);
}

```


图 5-19 getmaps 方法实现代码（2）

### 5.3.4 在线留言功能的实现

普通用户在首页菜单栏中点击在线留言按钮，在留言板页面可直接查看已有的留言信息。在 lyblist.jsp 中，先使用 sql 语句搜索出存入留言表的所有数据，然后调用

PageManager.getPages(url,4,sql,request)方法，返回一个 ArrayList 对象，接着在 for 循环里，使用 map.get()方法得到每个 ArrayList 对象的数据，然后放入模板里显示出来。

留言信息页如图 5-20 所示。

在线留言		我要留言
	留言于:2020-05-11 15:43:55.0 这个会议室预约系统用着挺方便，但是再把界面完善一点就更好了！	
	昵称: maining77    电话: 13649407174    邮 箱: 625026709@qq.com    QQ: 625026709	
	管理员回复:	
	留言于:2020-05-11 14:50:52.0 这个会议室预约系统用得很方便，感觉工作效率又高了一截！	
	昵称: lili98    电话: 13649407177    邮 箱: 1396965345@qq.com    QQ: 1396965345	
	管理员回复:	
	留言于:2020-05-11 14:50:52.0 这个会议室预约系统用得很方便，感觉工作效率又高了一截！	

共2条 第1/1页    首页    上页    下页    末页    1 ▼

图 5-20 留言信息页

获取留言板信息的实现代码如下图 5-21 所示。

```
<%  
String url = "lyblist.jsp?l=1";  
String sql = "select * from liuyanban order by id desc";  
//out.print(sql);  
ArrayList<HashMap> list = PageManager.getPages(url,4,sql,request );  
int i=0;  
for(HashMap map:list){  
i++;  
}%>
```

图 5-21 获取留言信息实现代码

登录的用户进入留言信息页看到已有的留言后，若要留言，则点击页面右上角的我要留言这个按钮，页面会立即跳转到 lyb.jsp 即留言填写页面，发布留言时系统方法会把用户填写的留言数据封装在 HashMap 中，然后调用 CommDAO 的 insert 方法将留言内容插入留言板。

下图 5-22 展示了留言填写页面。

在线留言		查看留言
昵称:	<input type="text" value="lili98"/>	
头像:		
QQ:	<input type="text" value="1396965345"/>	
邮箱:	<input type="text" value="1396965345@qq.com"/>	
电话:	<input type="text" value="13649407177"/>	
内容:	<div><div><p>这个会议室预约系统用得很方便，感觉工作效率又高了一截！</p></div><div>* 请输入留言内容</div></div>	
<div><input type="button" value="提交"/> <input type="button" value="重置"/></div>		

图 5-22 留言填写页面

在输入内容时，对于必须填写的内容，如用户名、留言内容，若空白不填写，

则无法成功留言，且会有红色文字弹出提示。主要实现代码如下图 5-23、图 5-24 所示。

```
<%  
HashMap ext = new HashMap();  
//ext.put("tglparentid", tglparentid);  
ext.put("huifuneirong", "");  
new CommDAO().insert(request, response, "liuyanban", ext, true, false, "lyblist.jsp");  
%>  
  
. . . . .
```

图 5-23 留言填写实现代码（1）

```
<script language=javascript >  
function checkform(){  
var chengobj = document.getElementById("cheng");  
if(chengobj.value==""){  
document.getElementById("clabelcheng").innerHTML="&nbsp;&nbsp;&nbsp;<font  
color=red>请输入昵称</font>";  
return false;  
}else {  
document.getElementById("clabelcheng").innerHTML=" ";  
}  
  
var neirongobj = document.getElementById("neirong");  
if(neirongobj.value==""){  
document.getElementById("clabelneirong").innerHTML="&nbsp;&nbsp;&nbsp;<font  
color=red>请输入留言内容</font>";  
return false;  
}else {  
document.getElementById("clabelneirong").innerHTML=" ";  
}  
return true;  
}  
</script>
```

图 5-24 留言填写实现代码（2）

### 5.3.5 系统用户管理功能的实现

管理员输入账号密码登录到系统后台，然后实现对系统的管理功能，即对各个模块功能进行增删改审核等。后台主页如图 5-25 所示。



图 5-25 后台主页

注册用户填写好信息之后，相关方法会把信息存入用户注册信息表中，因为用户注册之后还需要管理员审核通过才能登录，因此管理员需要进入系统用户管理界面审核注册的用户。若是未审核的，则表格中审核列为否，此时点击此按钮，确定执行操作，使否变为是，则审核成功。

注册用户管理页面如图 5-26 所示。



图 5-26 注册用户管理页

管理员审核的代码实现如下图 5-27 所示。

```
<%  
String yuan=request.getParameter("yuan");  
String id=request.getParameter("id");  
String tablename=request.getParameter("tablename");  
String sql="";  
if(yuan.equals("是"))  
{  
    sql="update "+tablename+" set issh='否' where id="+id;  
}  
else  
{  
    sql="update "+tablename+" set issh='是' where id="+id;  
}  
new CommDAO().commOper(sql);  
out.print("<script>location href='"+request.getHeader("Referer")+"'</script>");  
%>
```

图 5-27 管理员审核实现代码

### 5.3.6 站内新闻管理功能的实现

为了让系统的内容具有丰富性，还添加了日常性的新闻内容，主要是由管理员对新闻进行添加和修改的操作。添加新闻数据的页面采用了一个开源的在线 HTML 可视化编辑器 KindEditor，可以实现 word 文档中所见即所得的编辑效果，让新闻添加效果具有多样化。

添加新闻数据页面如图 5-28 所示。



添加新闻数据:


标题:	<input type="text" value="中国邮政首发战疫邮票"/> *
类别:	<input type="text" value="站内新闻"/> *
内容:	<div>  </div> <p>今天, 中国邮政特别发行1套《众志成城 抗击疫情》邮票, 军人、医护、科研人员、社区工作者、警察、志愿者的形象, 被以“众”字相连。</p> <p>据长江日报, 该邮票全套两枚, 面值各1.20元, 分别为: 众志成城、抗击疫情, 采用连票的形式, 以一个“众”字将两枚邮票紧密连接, 寓意着面对重大公共卫生事件, 在党中央坚强领导下, 全国上下凝聚起众志成城、抗击疫情的强大力量。计划发行1450万套, 邮票发行收入将捐</p>
添加人:	<input type="text" value="hsg"/> *
首页图片:	<input type="text" value="upload/1589210726821.jpg"/> <input type="button" value="上传"/>
点击率:	<input type="text" value="18"/> *
<input type="button" value="提交"/> <input type="button" value="重置"/>	

图 5-28 添加新闻数据页面

采用在线编辑器的实现代码如下图 5-29 所示。

```

<script>
KindEditor.ready(function(K) {
    var editor1 = K.create('textarea[name="neirong"]', {
        cssPath : 'kindeditor_a5/plugins/code/prettify.css',
        uploadJson : 'kindeditor_a5/jsp/upload_json.jsp',
        fileManagerJson : 'kindeditor_a5/jsp/file_manager_json.jsp',
        allowFileManager : true,
        afterCreate : function() {
            var self = this;
            K.ctrl(document, 13, function() {
                self.sync();
                document.forms['example'].submit();
            });
            K.ctrl(self.edit.doc, 13, function() {
                self.sync();
                document.forms['example'].submit();
            });
        }
    });
    prettyPrint();
});
</script>

```

图 5-29 在线编辑器实现代码

## 5.4 本章小结

本章主要介绍了本楼宇会议室预约系统的一些核心功能实现过程和实现方法，以及给出了实现界面截图和粘贴出一些核心代码。

## 第六章 系统测试

### 6.1 系统测试的目的及意义

完成对系统的分析设计以及编码后，为了确保系统运行环节不出错，就需要通过测试的手段对系统的各个环节部分进行测试操作，通过对应的测试手段去分析发现系统存在的不足和问题，检查程序是否能够正常运行，进行反复的测试和完善，避免系统在运行中后期出现缺陷。

### 6.2 系统功能性测试

#### 6.2.1 普通用户注册测试

对普通用户注册测试主要是对本系统的注册功能进行测试。包括注册填写信息不正确的错误反馈，注册成功后系统显示的提示等。测试结果如图 6-1 所示。

功能模块	方案	输入数据	预期结果	测试结果
普通用户注册测试	使用新用户名和完整的数据注册	用户名: xiaoli 密码: xiaoli55 确认密码: xiaoli55	系统提示注册成功	通过
	使用存在的用户名和完整的数据注册	用户名: xiaoli 密码: xiaoli55 确认密码: xiaoli55	系统提示注册失败，用户名已存在	通过
	使用未注册的用户名和完整的数据注册，但确认密码不一样	用户名: shushu555 密码: 19980205 确认密码: 199802	系统提示注册失败，两次输入密码不一样	通过

图 6-1 用户注册测试

普通用户注册页面实现图如图 6-2 所示。



The image shows a 'User Registration' form with the following fields: Username (required), Password (required), Confirm Password, Name (required), Gender (dropdown menu with 'Male' selected), Birth Date, QQ, Email, Mobile Phone, ID Card, Avatar (with an 'Upload' button), Address, Remarks (text area), and 'Submit'/'Reset' buttons at the bottom.

图 6-2 普通用户注册页面图

## 6.2.2 普通用户登录测试

登录测试是对本系统的登录功能进行测试。即登录异常的信息反馈，登录成功后页面导航菜单的自动加载等。测试结果如图 6-3 所示。

功能模块	方案	输入数据	预期结果	测试结果
普通用户登录测试	使用注册成功的用户名、密码、页面显示的验证码登录	用户名: <u>xiaoli</u> 密码: xiaoli55 验证码: 1314	系统提示登录成功, 页面跳转到前台主页	通过
	使用注册成功的用户名、密码、页面显示的验证码登录	用户名: <u>xiaoli</u> 密码: 19981 确认密码: 1314	系统提示登录失败, 显示密码错误	通过
	使用存在的用户名、密码、和页面显示的验证码不同的验证码登录	用户名: <u>xiaoli</u> 密码: xiaoli55 确认密码: 8733	系统提示登录失败, 显示验证码错误	通过

图 6-3 普通用户登录测试

普通用户登录页面实现图如图 6-4 所示。

图 6-4 普通用户登录页面图

### 6.2.3 会议室预约测试

会议室预约测试是对本系统的会议室预约功能进行测试。测试结果如图 6-5 所示。

功能模块	方案	输入数据	预期结果	测试结果
会议室预约测试	普通用户登录成功，进入预约界面，填写预约人、选择预约时间、预约时段等信息	预约人：xiaoli 预约时间：2020-05-15 预约时段：10:00-11:30	系统提示会议室预约操作成功	通过
	普通用户登录成功，进入预约界面，没有填写预约人、选择预约时间、预约时段等信息	预约人： 预约时间： 预约时段：	系统提示预约操作失败	通过

图 6-5 会议室预约测试表

会议室预约页面实现图如图 6-6 所示。

会议室预约

会议室编号:

0004

会议室名称:

会议室4

会议室类型:

A类

会议室面积:

2500

预约时间:

预约时段:

8:30-10:00

预约人:

lili98

备注:

提交

重置

图 6-6 会议室预约页面图

## 6.2.4 密码修改测试

密码修改测试是对系统用户管理中的密码修改功能进行测试。测试结果如图 6-7 所示。

功能模块	方案	输入数据	预期结果	测试结果
密码修改测试	用户登录，输入 1 次原密码、2 次新密码，点击提交按钮	原密码：xiaoli55 新密码：xiaoli88 确认新密码：xiaoli88	系统提示面修改成功	通过
	原密码输入错误，2 次新密码输入相同	原密码：xiaoli 新密码：xiaoli88 确认新密码：xiaoli88	系统提示密码修改失败，原密码输入错误	通过
	原密码输入正确，2 次新密码输入不相同	原密码：xiaoli55 新密码：xiaoli88 确认新密码：xiaoli	系统提示密码修改失败，2 次新密码不同	通过

图 6-7 密码修改测试

密码修改页面实际效果图如图 6-8 所示。

修改密码

原密码:

新密码:

确认密码:

确定

重置

图 6-8 密码修改页面

### 6.2.5 在线留言测试

在线留言测试是对楼宇会议室预约系统的在线留言功能进行测试。测试结果如图 6-9 所示。

功能模块	方案	输入数据	预期结果	测试结果
在线留言测试	用户登录成功，进入留言界面，填写留言标题、内容等信息	留言标题：系统评价 留言内容：系统很好用	系统提示留言成功，显示新的留言信息	通过
	用户登录成功，进入留言界面，不填写留言标题、内容等信息	留言标题： 留言内容：	系统提示留言失败，显示内容不能留空白	通过

图 6-9 在线留言测试


在线留言页面实际效果图如图 6-10 所示。


在线留言


昵称：


\*


头像：











QQ：

邮箱：

电话：

内容：

\*

提交

重置

图 6-10 在线留言页面

### 6.3 本章小结

本章简单的介绍了楼宇会议室预约系统的测试。主要对系统的注册登录模块、

会议室预约模块、密码修改、在线留言模块进行功能性测试，介绍了测试的具体操作流程，还作出测试方案以及页面效果图。

## 第七章 总结

本文在通过对会议室预约系统的相关文献的调查，以及在系统用户的需求分析的基础上，运用 MVC 框架、B/S 架构、JSP 等技术实现了一个楼宇会议室预约系统。在此对本次毕业设计论文所进行的工作做一个总结。

工作前期主要是搜索相关文献进行调查与分析，然后结合传统人工管理模式的弊端，对目前企业在会议室预约方面留存的问题进行分析以及用户的需求分析，分析出需要实现的功能模块，然后结合业务流程图、数据流程图、数据字典等来实现对系统的进一步分析和设计，对用到的数据库表进行详细设计，接着在实现模块对系统的每个功能模块的实现流程和方法作出详细阐述，展示了相关的页面实现截图和代码截图。在对系统的分析设计过程中，主要使用了结构化系统开发方法和 MVC 框架、B/S 架构、JSP 等技术，程序编写完整，功能基本实现。

这次的毕设项目让我认识到了知识量对一个从事程序开发人员的重要性，当遇到问题的时候，需要寻找知道解决办法，就必须要将该问题领域的技术融会贯通，只有在自己学习掌握的技术知识足够充分，才能在开发中尽可能的少于难题。对于设计的此系统，也还存在着一些不足，还有一些功能需要完善。在接下来的学习工作生涯中，我将不断提升自己的技术水平，争取早日开发出一个真正意义上完美的项目。

## 参考文献

- [1] 王岩,贡正仙.数据库原理、应用与实践[M].清华大学出版社,2016.
- [2] 马鸿健,张耕云,王关祥,等.高校会议室系统设计与实现[J].科技视界,2016,17:100-101.
- [3] 包琦琦,郑梁梦.会议场馆预约管理系统的设计与实现[J].科技视界,2015,18:69.
- [4] 于洋.会议室预定系统的设计与实现[D].长春:吉林大学,2012.
- [5] 张津.基于 Struts 的会议室调度系统的设计与实现[D].成都:电子科技大学,2013.
- [6] 尚玉梅.高校办公室预约管理软件设计与实现[D].成都:电子科技大学,2014.
- [7] 孟凡亮,文福安.基于 J2EE 的会议室预约系统设计与实现[J].中国电子商情·通信市场,2010,6:45-48.
- [8] 高翔.基于 Struts 框架的会议室预约管理系统设计与实现[D].青岛:中国海洋大学,2008.
- [9] 明日科技. Java Web 从入门到精通[M]. 清华大学出版社, 2019.
- [10] 黄丹.会议室预约系统的设计与实现[J].电子技术与软件工程,2014,4:72.
- [11] 杨超,雷龙涛.基于 .net 的会议室预约系统的设计与实现[J].信息通信,2015,11:103-104.
- [12] 陈贺婉,杨广军.基于 WEB 的高校会议室管理系统的设计与实现[J].教育信息化,2006,11:44-46.
- [13] 郑无寒.基于 B/S 架构的企业会议管理平台的设计与实现[D].西安:西安电子科技大学,2015.
- [14] 周光前.基于 B/S 模式小型酒店住宿预订系统设计与实现[D].南京:东南大学,2015.
- [15] 张雪敏.浅议 MVC 设计模糊在 JavaWeb 中的作用[J].电子信息,2018,22:58-59.
- [16] 于晗.运用 JDBC 技术连接和操作 MySQL 数据库的方法[J].林区教学,2014,9:100-103.
- [17] 周益平.基于 JSP 技术的协同 OA 系统设计与开发[D].大连:大连理工大学,2016.
- [18] 李冬云,赵亦松.基于 MVC 模式的会议管理系统的设计[J].北京联合大学学报(自然科学版),2008,4:34-39.
- [19] 全俊松.企业内部会议及会议室信息化管理系统设计与开发[D].天津:天津大学

软件学院,2016.

[20] Ankur Bawiskar, Prashant Sawant.Integration of Struts, Spring and Hibernate for an University Management System[J].International Journal of Emerging Technology and Advanced Engineering,2012,6:203-210.



## 致 谢

本次毕业设计是在凌兴宏老师的指导帮助下完成的。凌老师为人谦虚、知识储备量极大且尽职尽责，在我的毕业设计完成期间，给予了我很多的帮助。他会不时的给予我一些建议性意见，在我遇到难以解决的问题时，向他咨询，他总能很耐心并正确的指出我的错误思路并给予纠正。在我撰写相关文档期间，他也能仔细的查看其中问题并及时的给予纠正，用他的耐心和细心给我提出了很多细节上的错误，让我能够更好的完成毕业设计相关任务。

在这次的毕业设计过程中，许多同学也给予了我很多的帮助，在项目设计上也有给我提出了一些好主意，在平时的交流中也给予了我一些启示，并在一些细节上给我及时纠正，让我少走了很多弯路。在此，对帮助过我的老师与同学们表示感谢。