**教室、会议预约管理系统**

**总体设计分析报告**

**软件名称**： 会议室、教室管理系统

**学生姓名**： 马瑄

**专 业**： 计算机科学与技术

**班 级**： 计1402

**学 号**： 41455030

**组 号**： 16

**目录**

[1 引言 5](#_Toc16043523)

[1.1 编写目的 5](#_Toc16043524)

[1.2 背景 5](#_Toc16043525)

[1.3 定义 5](#_Toc16043526)

[1.4 参考资料 5](#_Toc16043527)

[2 项目概述 5](#_Toc16043523)

[2.1 需求概述 5](#_Toc16043529)

[2.2 条件与限制 6](#_Toc16043530)

[2.3 运行环境 6](#_Toc16043531)

[3 总体设计 6](#_Toc16043533)

[3.1 系统处理流程 6](#_Toc16043534)

[3.2 C/S端设计 7](#_Toc16043535)

[3.3 模块处理 10](#_Toc16043540)

[4 接口设计 10](#_Toc16043533)

[4.1 用户接口 11](#_Toc16043540)

[4.2 外部接口 11](#_Toc16043540)

[5 数据结构设计 12](#_Toc16043533)

[5.1 概念结构设计 12](#_Toc16043540)

[5.2 逻辑结构设计 13](#_Toc16043540)

[6 运行设计 15](#_Toc16043533)

[6.1 运行模块组合 15](#_Toc16043540)

[6.2 运行控制 15](#_Toc16043540)

[6.3 运行时间 15](#_Toc16043540)

[7 系统出错处理设计 16](#_Toc16043533)

[8 测试计划 16](#_Toc16043533)

1. **引言**

**1.1编写目的**

确定系统的物理模型和各功能的逻辑实现方案，进一步细化系统结构，为编程人员和设计人员提供项目交流的工具。选定最佳方案和最合理的软件结构，从而用较低的成本开发出较高质量的软件结构。

**1.2背景**

项目名称：会议教室管理系统

项目任务开发成员：马瑄、任琳琳、宛嵇祥、张菁彧

项目任务用户：管理员、用户

项目业务范围：包括学校可租借教室会议室及其相关设备的管理、预订服务。

**1.3定义**

静态数据－－系统固化在内的描述系统实现功能的一部分数据。

动态数据－－在软件运行过程中用户输入的后系统输出给用户的一部分数据，也就是系统要处理的数据。

C/S架构——又称主从式架构或客户端-服务器结构（Client/Server），服务器被动地接收来自客户端的请求，处理请求并传回结果。

**1.4参考资料**

<<软件工程导论>>（第六版）　 ------张海藩 编著 清华大学出版社

<<软件工程>>（第九版）------张海藩编著人民邮电出版

1. **项目概述**

**2.1需求概述**

1) 登录功能：验证登录用户是否为数据库中的合法用户，判断登录的用户的身份，一般用户只能实现教室/会议室的浏览、查看、搜索、预定；管理员可以对用户信息进行更改，审核预约信息。

2) 查询功能：可以浏览教室/会议室的占用情况，也可以查找具体教室的相关信息。用户通过这个功能，可以借教室，即获得教室在某段时间的使用权，办讲座，开展社团活动

3) 预约功能：用户通过搜索或浏览虚招到合适的教室进行预约。长期预约，以及对长期预约过程中可能存在冲突的解决办法，比如审批和抽签。

4) 用户管理功能：管理员可以添加新的用户以及修改登录用户的密码，可以修改用户的相关信息。

5）自习功能：用户查询当天某一时段或多个时段的空闲教室，去上自习。

6)审核功能（待定）：管理员对于用户的预约进行审核，以优先级解决冲突问题（抽签等）

**2.2条件与限制**

由于编程人员和开发时间的限制，系统功能实现简单，由于成员信息条目小，只能对部分信息进行统计，功能上有所欠缺，不能完全满足当前用户对信息管理的需要。本系统可嵌入其他信息系统，共用同一数据库，不同的用户有其不同的权限，这样更能提高系统的稳定性和完善系统的功能，同时，也更加满足用户的需求。

a． 所建议系统的运行寿命的最小为3年；

b． 进行系统方案选择比较的时间为2个星期；

c． 硬件、软件、运行环境和开发环境方面的条件和限制；

d． 系统投入使用的最晚时间为2017年8月1日。

2.3运行环境

客户端操作系统：Windows10

服务器数据库：sqlsever

服务器操作系统：window10

编程软件：IntelliJIDEA

1. **总体设计**

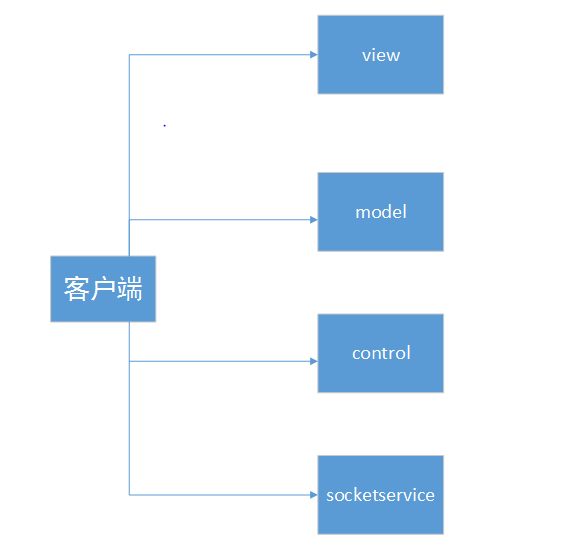
**3.1系统处理流程**

系统的处理流程如下：用户登录系统进行身份验证，学生（或教师）可以查询、预定空闲教室。管理员根据用户的申请对信息进行审核处理。具体流程图如下：



**3.2 C/S端设计**

客户端（client）设计如下：



注释：

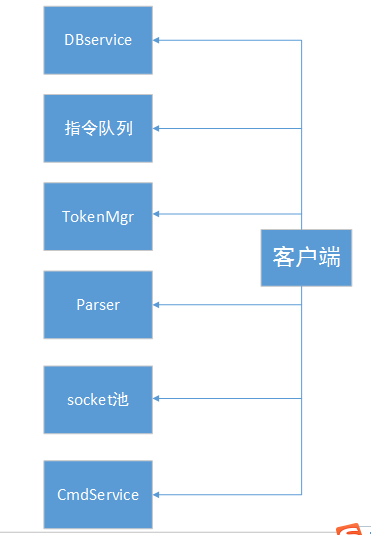
View：处理数据显示的部分,通常视图是依据模型数据创建的。

model：处理应用程序数据逻辑的部分，通常模型对象负责在数据库中存取数据。

Control：是应用程序中处理用户交互的部分。控制器负责从视图读取数据，控制用户输入，并向系统发送数据。

Socketservice:客户端使用的socket连接服务,使用者可以向它发送字符串,保证将这些字符串以加密形式发送到目标IP。

服务器端（service）设计如下：



注释：

DBService:服务器端使用的数据库接口。根据本软件的需要对基本的MySQL操作进行封装，调用对应函数，得到相应的对象。

指令队列：服务端为了简化数据库操作逻辑，使用指令队列实现数据库操作的单线程化。所有的关于数据库的操作都经过指令队列代理，提高了效率。

TokenMgr:一个客户端上线并成功登录后，服务端会分配给该客户端一个token作为标识，客户端需要提供token作为执行其它操作的凭证。客户端下线后，token被回收。通过token可以实现数据加密，一定程度上实现系统的安全性。

Parser：服务器的解析模块，解析来自token池的消息，验证连接的token和操作是否匹配，同时也可以调度指令队列。

Sockect池：用于监听特定端口，对于来自这一端口的连接尝试，首先将保存连接实例，并与之交换公钥以建立安全连接。建立安全连接后，将从这一端口的数据发送给Parser。

CmdService：命令服务。客户端和服务端都具有该服务，但是使用的侧重点不同，在服务器中充当两种对象的转换。

**3.3模块处理**

教室、会议室管理系统一级分解



教室、会议室管理系统二级分解



1. **接口设计**

**4.1用户接口**

在用户界面部分，要尽量设计出友好、高效的人机界面。要注意界面的布局，突出显示重要和出错的信息。总的来说，系统的用户界面应做到可靠性、简单性、易学习和使用。

管理员

教师

学生

用户：

密码：

登陆

教室、会议室管理系统

**4.2外部接口**

硬件接口：一般的个人计算机；

软件接口：Windows10，sqlsever进行数据库命令和保存。

**4.3内部接口设计**

操作模块如下：

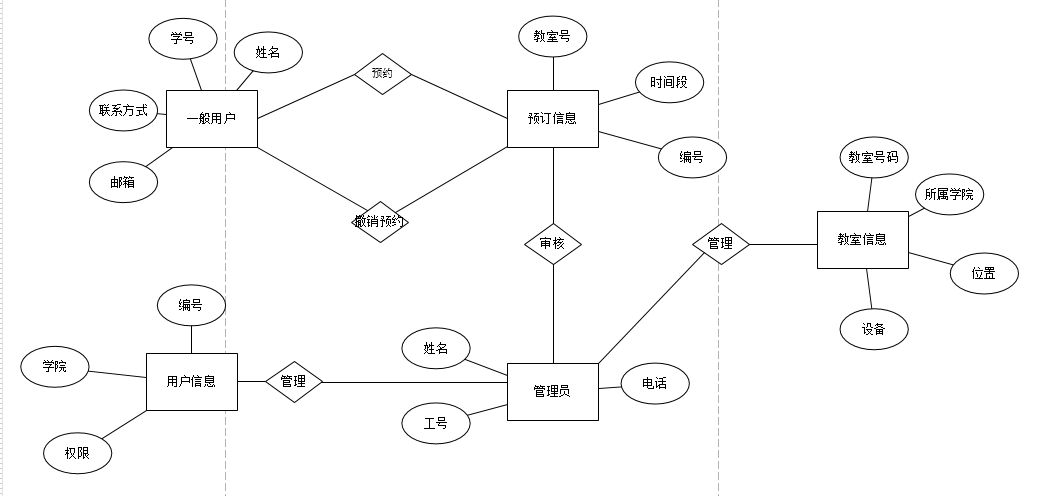


1. **数据结构设计**

由于该系统数据过于庞大，利用数据库来管理这些数据，可以方便进行查询和更新。

**5.1概念结构设计**

由各个实体间的练习及其各属性的关系，可得到E-R图：



**5.2逻辑结构设计**

可得到关系数据库表如下

用户表；

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 备注 |
| 学号/工号 | varchar（10） | 主键 |
| 姓名 | varchar（10） |  |
| 学院 | varchar（20） |  |
| 班级 | varchar（10） | 教师为NULL |
| 电话 | varchar（20） |  |
| 邮箱 | varchar（20） |  |
| 是否为学生 | Bool |  |

管理员表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 备注 |
| 工号 | varchar（10） | 主键 |
| 姓名 | varchar（20） |  |
| 学院 | varchar（20） |  |
| 邮箱 | varchar（20） |  |
| 电话 | varchar（20） |  |

教室、会议室表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 备注 |
| 教室编号 | varchar（10） | 主键 |
| 学院 | varchar（20） |  |
| 位置 | varchar（10） |  |
| 权限 | Int | 见下述注释 |
| 设备 | varchar（20） |  |
| 开放时间 | Date |  |
| 结束时间 | Date |  |

注释：权限说明：1（本院学生）；2（全部学生）；3（教师）

申请表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 备注 |
| 编号 | varchar（10） | 主键 |
| 申请人学号、工号 | varchar（10） |  |
| 教室编号 | varchar（10） |  |
| 开始时间 | Date |  |
| 结束时间 | Date |  |
| 申请理由 | varchar（30） |  |
| 状态 | Int | 见注释 |

注释：状态：1（审核通过）；2（等待审核）；3（审核未通过）。

1. **运行设计**

**6.1运行模块组合**

客户端程序输入相应的用户名和密码，建立sockect连接，请求一个公钥，客户端接受公钥后，从客户端发送公钥给服务器，实现用户与服务器的连接，登陆系统。中间调用了各个模块，产生相应的输出。

服务器端程序等待客户端建立连接，客户端请求公钥，服务器发送公钥并接受从客户端发送的公钥，从而登陆系统。其中，服务器端必须始终处于活动状态，以便接收数据返回给客户端。

**6.2运行控制**

运行控制将严格按照各模块之间的函数调用关系来实现。在各事务中心模块，需对运行控制进行正确的判断，选择正确的运行控制路径。

在连接方面，客户端请求一个公钥，服务器端接收信息并发送给客户端一个公钥，客户端接收公钥后，对公钥进行加密处理并发送给服务器端。服务器端解密公钥，核对信息并进行连接。

**6.3运行时间**

在软件的需求分析中，对运行时间的要求为必须对作出的操作有较快反应。影响时间的主要因素有c/s端口连接和服务器的性能。我们需要尽量减少网络连接的开销，提高效率。同时服务器的效率受数据库的影响，必须在保证数据完整的情况下提高效率，所以尽量使用高性能服务器，以减少操作反应时间。

1. **系统出错处理设计**

|  |  |  |
| --- | --- | --- |
| 系统输出信息的形式 | 含义 | 处理方法 |
| 账号登录异常 | 账号未注册或是被冻结 | 重新注册，申请解冻 |
| 数据库连接不上 | 由于网路堵塞繁忙，数据库软件繁忙，连接数据库配置不正确等一些因素 | 等待连接、修复网络、更改数据库配置 |
| 预定受限 | 对于教室没有预定权限 | 更换用户、申请获取权限 |
| 重复操作 | 由于网络阻塞延时，人为的多次提交导致信息的反复录入 | 等待界面反应、刷新界面 |

1. **测试计划**

测试过程预计分为3个阶段：

a）单元测试阶段，此阶段主要进行各个模块的内部测试工作。从数据接口、数据结构和边界条件等方面进行测试。主要采用白盒测试的技术，可以对多个模块进行并行测试。

b）集成测试阶段，本阶段进行模块之间配合工作的测试，验证各个模块配合工作的稳定性是否符合预期。把模块按照要求组装起来进行测试，主要检查与接口有关的问题。

c)验收测试/平行运行阶段：这一阶段将本系统投入模拟使用的环境，比较实际运行的结果和预期是否一致，是否符合用户需求。