**一、实验目的**

通过本课程，进一步复习、巩固所学的数据结构与算法、计算机系统基础、密码学基础、社会科学中的计算思维方法等课程的基本概念、技术原理、软件设计的方法与技术等，初步掌握软件设计的流程与基本方法，能够运用C/C++、Java、Python等至少一种编程语言进行软件开发。

**二、实验内容**

会议室管理与预约系统：设计与开发一个会议室预订与管理系统，功能包括但不限于：支持管理员登录功能、 支持对会议室资源进行管理，会议室资源包括会议室名称、容纳人数等；支持对会议室进行预约管理，从满足需求的可用会议室资源中给出预约建议，能够对会议室 进行预约/取消预约

1. **实验工具和环境**

实验工具：visual studio code、phpstudy、chrome

实验环境：Windows10、LINUX:apache、msql、php

使用语言：HTML、JavaScript、PHP

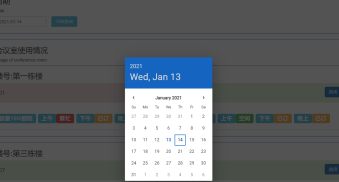
1. **软件功能设计（用文字和示意图说明各个功能点）**
2. 登录窗口功能（login.html）：

登录类型有两种，一种是普通用户的权限有预约，退订的功能；管理员的权限则有取消任何预定以及设置不可预订，增加删减会议室。

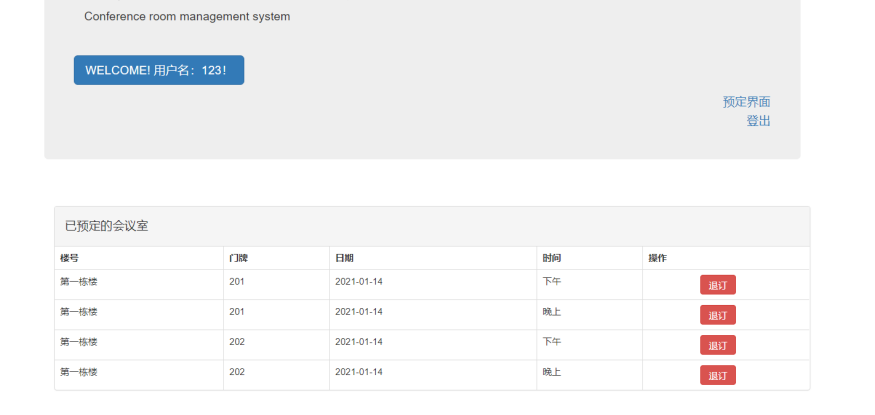
1. 会议室的管理员功能(admin.html)：

该界面可以对会议室资源进行综合管理，该界面直接连接数据库，修改的内容等于修改数据的内容，可以做到随时各个界面的数据共享。增加会议室建筑楼号、删除会议室建筑楼号，增加、删除、修改会议室的各项属性。

1. 普通用户的预约功能(index.html)：

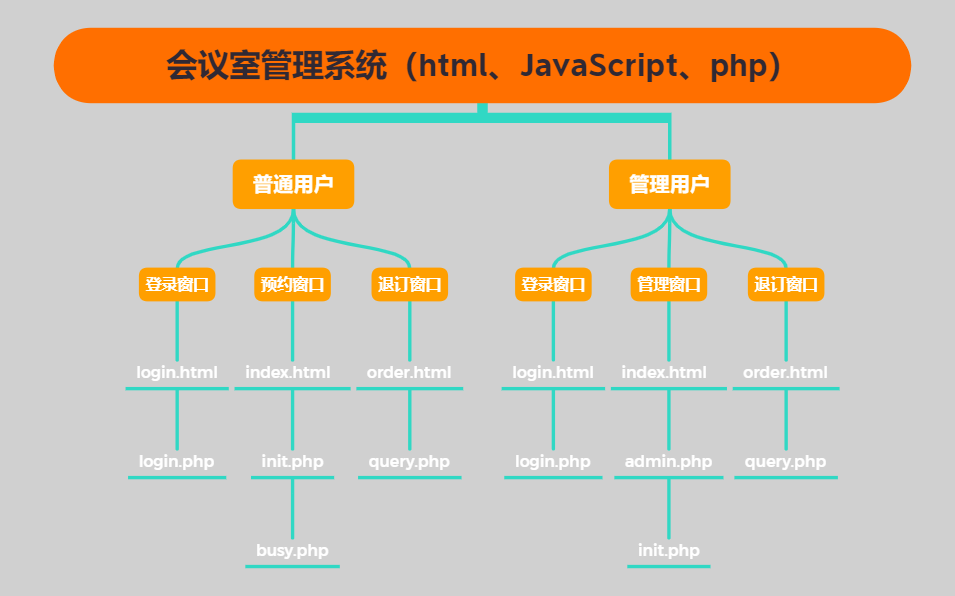
根据从数据库加载出来的数据，来填充页面。先使用时间查询

查出这个日期的时间的会议室状况，然后用户根据查出来的情况进行自主的会议室选择，如果是繁忙代表着已经被其他人预定或者是管理员禁用，已订代表着已经预定成功。如果时间各个时间段已经被订满会将显示框变为红色，已满点击也无法预定。

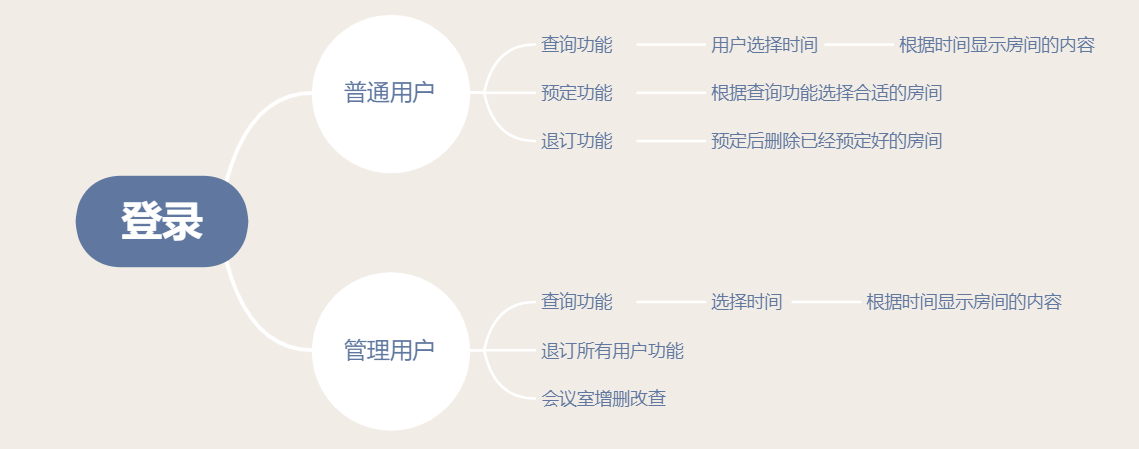
1. 用户的退订以及管理员可以取消任何人的预定(order.html)：

点击退订即可删除数据库的预定内容

1. **软件结构设计（用文字和模块结构图说明软件的组成和结构，以及模块间的函数接口）**



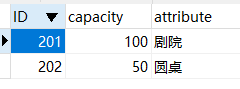
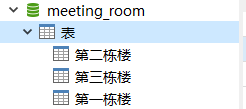
1. login.html：使用ajax里的post请求收集用户名以及密码等程序与后端程序login.php进行交互，后端返回OK则代表数据库有该用户，FALSE则代表用户不存在，登录失败­­。
2. login.php：拿到login.html传过来的参数username以及password，使用select查询语句查询数据库usr中的usr表以及admin表的比对，返回结果。
3. index.html：主要的功能界面，先使用ajax里的post请求init.php界面拿到所有会议室的记录，先进行第一次渲染，再使用ajax里的post请求busy.php拿到已被预定的会议室，进行用户分类再对界面进行填充。
4. init.php：使用select语句查询数据库meeting\_room，先遍历所有的表，每一张表对应一栋会议楼，再分别遍历每张表去拿到每个会议室的内容，输出。
5. busy.php：使用select语句查询数据库using中的roomusing表，其中记录了所有的预定情况，遍历表中的所有内容并且输出。
6. admin.php：管理界面功能模块主要的后端脚本，该模块主要是负责管理会议室的所有内容，使用mysql语句对数据库中的meeting\_room所有表进行增、删、改、查等基础操作
7. query.php：实现对表roomusing的维护，主要是实现会议室的退订功能。
8. **软件运行流程设计（用文字和流程图说明软件实现各个功能的运行流程）**



1. 进入登录界面login.html，普通用户登录成功后跳转到index.html，该界面显示出所有会议室的情况，然后用户根据自身需求，选择合适的时间，然后数据根据时间渲染出界面，用户选择合适时间，载入数据库。点击退订然后退订界面有退订的功能区操作数据库
2. 管理用户登录login.html后，跳转到index.html,该界面会显示出会议室的情况，管理者可以根据该界面调整所需要的会议室。也可以对会议室的所有预定进行退订，最后对会议室的内容进行维护。
3. **关键模块设计（重要的核心模块内部的算法、处理流程和实现方法，可以贴少部分核心函数，包括模块内部关键函数之间的调用关系）**
4. 前端会议室查看初始化代码(index.html)
5. $.post("init.php",
6. {
7. name:"init",
8. },**function**(data,status)
9. {
10. //防止重复点清空
11. $('#all\_panel').html('');
12. **var** parsedJson = jQuery.parseJSON(data);
13. **var** building=parsedJson[0].building;
14. **var** pre\_first="<div class=\"panel panel-primary\"  class=\"container-fluid\" id="+building+" ><div class=\"col-md-12\"><h3>"+"楼号:"+building+"</h3></div></div> ";
15. $('#all\_panel').append(pre\_first);
16. **for**(**var** i =0;i<parsedJson.length;i=i+2)
17. {
18. **if**(building!=parsedJson[i].building)
19. {
20. building=parsedJson[i].building;
21. **var** pre\_first="<div class=\"panel panel-primary\"  class=\"container-fluid\" id="+building+" ><div class=\"col-md-12\"><h3>"+"楼号:"+building+"</h3></div></div> ";
22. $('#all\_panel').append(pre\_first);
23. }
24. //追加教室 first 门牌号
25. //这里是first内容
26. $('#'+building).append(first);
27. }
28. });

以上模块是用来初始化会议室的所有属性并且存在#all\_panel该标签中，该代码通过AJAX请求init.php的模块

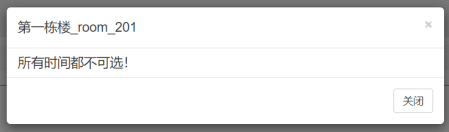
1. 前段会议室查看初始化代码的后端交互代码(init.php)
2. <!-- /\*
3. \*author:gzhu\_ruiop
4. \*date：2021-01-13
5. \*des：会议管理系统···初始化模块读取会议室meeting\_room所有信息
6. \*/ -->
7. <?php
8. header("Content-Type: text/html;charset=utf-8");
9. $con = mysqli\_connect("47.96.151.142","root","110124");
10. $select\_db = mysqli\_select\_db($con,'meeting\_room');
11. **if** (!$select\_db)
12. {
13. **die**("could not connect to the db:\n" .  mysqli\_error());
14. }
15. mysqli\_query($con,"set names 'utf8'");
16. $result = mysqli\_query($con,"SHOW TABLES");
17. $arry=**array**();
18. **while**($row = mysqli\_fetch\_array($result))
19. {
20. //这里的row(0)代表每个元素
21. //查询代码
22. $sql = "select \* from ".$row[0];
23. $building=$row[0];
24. mysqli\_query($con,"set names 'utf8'");
25. $res = mysqli\_query($con,$sql);
26. **if** (!$res)
27. {
28. **continue**;
29. **die**("could get the res:\n" . mysqli\_error());
30. }
31. **while** ($row = mysqli\_fetch\_assoc($res))
32. {
33. $row['building']=$building;
34. $arry[]=$row;
35. }
36. //关闭数据库连接
37. }
38. echo json\_encode($arry);
39. mysqli\_close($con);
40. ?>

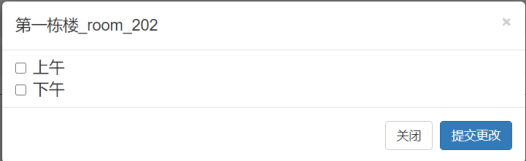
该模块请求数据库的表项

从左往右是会议室ID、容量、属性

1. 初始化后填充已被预定的房间代码(index.html)
2. $.post("busy.php",
3. {
4. time:select\_time,
5. },
6. **function**(data,status)
7. {
8. console.log("busy测试成功"+select\_time);
9. **var** parsedJson = jQuery.parseJSON(data);
10. **for**(**var** i =0;i<parsedJson.length;i++)
11. {
12. **var** id\_building='#'+parsedJson[i].building;
13. **var** id\_room=id\_building+'\_room\_'+parsedJson[i].roomnum;
14. **var** using\_time\_num=parsedJson[i].usingtime;
15. **var** sql\_username=parsedJson[i].username;
16. **var** id\_using\_time;
17. **if**(using\_time\_num==1)
18. {
19. id\_using\_time="#morning";
20. }**else** **if**(using\_time\_num==2)
21. {
22. id\_using\_time="#afternoon";
23. }**else**
24. {
25. id\_using\_time="#evening";
26. }
28. **if**(sql\_username==username)
29. {
30. $(id\_room).children().find(id\_using\_time).html("已订");
31. $(id\_room).children().find(id\_using\_time).removeClass("label-success").addClass("label-warning");
32. }**else**
33. {
34. $(id\_room).children().find(id\_using\_time).html("繁忙");
35. $(id\_room).children().find(id\_using\_time).removeClass("label-success").addClass("label-danger");
36. }
37. console.log("id\_room:"+id\_room+"id\_using\_time:"+id\_using\_time);
39. //如果三个标签都为繁忙 那么整个格子变红 并且按钮为禁用
40. **if**(($(id\_room).children().find('#morning').hasClass("label-danger")||$(id\_room).children().find('#morning').hasClass("label-warning"))&&($(id\_room).children().find('#afternoon').hasClass("label-danger")||$(id\_room).children().find('#afternoon').hasClass("label-warning"))&&($(id\_room).children().find('#evening').hasClass("label-danger")||$(id\_room).children().find('#evening').hasClass("label-warning")))
41. {
42. $(id\_room).removeClass("panel-success").addClass("panel-danger");
43. $(id\_room).children().find('.btn').removeClass("btn-primary").addClass("btn-danger");
44. $(id\_room).children().find('.btn').html("已满");
45. }
46. }
47. });

请求busy.php，请求的参数是选中的日期，根据选中的日期拿到已经预定的日期，根据已经预定的日期选中记录，根据记录中保存的用户名进行分类，如果刚好是登录的用户名，根据用户名分类“繁忙”或“已订”。“繁忙”代表着已经被其他人预定，“已订”代表着已经被该用户预定

如果该房间的时间段都被预定，则将该标签框置红，并且操作按钮设置“已满”，点击将不可操作，不可操作将弹出alert提示框。

如果可以进行选用则会弹出可以选用的时间段

提交更改则通过add.php后端脚本维护数据库

1. 添加房间后端模块代码(add.php)
2. <!-- /\*
3. \*author:gzhu\_ruiop
4. \*date：2021-01-13
5. \*des：会议管理系统···添加预定的房间模块
6. \*/ -->
7. <?php
8. $time=$\_POST['time'];
9. $username=$\_POST['username'];
10. $date=$\_POST['date'];
11. $building=$\_POST['building'];
12. $room=$\_POST['room'];
13. $time\_arry = explode(',',$time);
15. header("Content-Type: text/html;charset=utf-8");
16. $con = mysqli\_connect("47.96.151.142","root","110124");
17. $select\_db = mysqli\_select\_db($con,'using');
18. **if** (!$select\_db)
19. {
20. die("could not connect to the db:\n" .  mysqli\_error());
21. }
22. //时间数组，如果是morning 1
23. $time\_num=array();
24. **for**($index=0;$index<count($time\_arry);$index++)
25. {
26. **if**($time\_arry[$index]=="morning")
27. {
28. $time\_num[]=1;
29. }**else** **if**($time\_arry[$index]=="afternoon")
30. {
31. $time\_num[]=2;
32. }**else** **if**($time\_arry[$index]=="evening")
33. {
34. $time\_num[]=3;
35. }
36. }
37. **for**($index=0;$index<count($time\_num);$index++)
38. {
39. $sql = "INSERT INTO roomusing ".
40. "(ID,roomnum, building,usingtime,usingdate,username) ".
41. "VALUES ".
42. "(NULL,'$room','$building','$time\_num[$index]','$date','$username')";
43. mysqli\_query($con,"set names 'utf8'");
44. $res = mysqli\_query($con,$sql);
45. }
46. echo "OK";



添加该数据库的表项从左到右是ID号、门牌号、楼号、使用日期、使用时间、预定的用户

1. 退订房间前端代码模块(order.html)
2. **function** click\_func(elem)
3. {
4. console.log(elem.id);
5. $.post('query.php',
6. {
7. style:"delete",
8. id:elem.id,
9. },**function**(result){
10. **if**(result[0]== 'O')
11. {
12. alert("删除成功");
13. window.location.reload();
14. }
15. });
16. }



点击退订即可通过ajax中的post中请求数据库中删除已预定的记录

1. 查询预定记录、退订房间后端代码模块(query.php)
2. #根据传入的style决定哪种方式 查询还是删除
3. **if**($style=="find")
4. {
5. //查询代码
6. $sql = "select \* from roomusing where username='$username'";
7. //查询使用utf-8编码方式
8. mysqli\_query($con,"set names 'utf8'");
9. //使用sql语句进行查询
10. $res = mysqli\_query($con,$sql);
11. //查询失败结束进程
12. **if** (!$res)
13. {
14. die("could get the res:\n" . mysqli\_error());
15. }
16. //创建一个数组拿到每一行的数据
17. $arry=array();
18. **while** ($row = mysqli\_fetch\_assoc($res))
19. {
20. $arry[]=$row;
21. }
23. //将数组最后以json格式输出
24. echo json\_encode($arry);
25. //关闭数据库连接
26. }**else** **if**($style=="delete"){
27. //通过post拿到删除的id号
28. $ID=$\_POST['id'];
29. //删除语句
30. $sql="DELETE FROM roomusing WHERE ID=$ID";
31. //设置utf8编码
32. mysqli\_query($con,"set names 'utf8'");
33. //执行sql语句
34. $res = mysqli\_query($con,$sql);
35. **if** (!$res) {
36. die("could get the res:\n" . mysqli\_error());
37. }**else**{
38. echo "OK";
39. }
40. }

通过post请求读取其中查询、删除的方式，如果是删除则拿到要删除的id，进行数据库的操作

1. 管理房间的前端模块(admin.html)
2. **var** building=Array();//该数组是保存所有从数组库读出来的建筑楼号
3. **var** str\_select\_building=**null** //该变量是保存选中的建筑楼号，传给添加到get后进行该建筑的遍历

­上面两个变量是该文件中的全局变量，主要是保存第一次从数据库中读出的会议室的总楼数组，以及目前页面选中的楼号，通过目前选中的楼号可以遍历出该会议楼所有会议室的状况，方便对该会议室的属性进行维护，每栋会议楼对应一张表

1. **function** find\_tables()
2. {
3. building=Array();
4. $.get('admin.php',
5. {
6. style:"find\_tables",
7. },**function**(results)
8. {
9. //console.log(results);
10. //先遍历完拿到多少building
11. **var** json=jQuery.parseJSON(results);
12. //console.log(json);
13. **for**(**var** i=0;i<json.length;i++){
14. building.push(json[i].Tables\_in\_meeting\_room);
15. }
16. **var** li\_str="";
17. **for**(**var** i=0;i<building.length;i++)
18. {
19. //console.log(building[i]);
20. //这里拼一个带参改 href=temp.html?building=first;
21. li\_str=li\_str+"<li><a href=\"admin.html?building="+escape(building[i])+"\">"+building[i]+"</a></li>";
22. }
23. li\_str=li\_str+"<button class=\"btn btn-default \"  data-toggle = \"modal\"  data-target=\"#add\_building\_modal\">增加楼号</button>";
24. li\_str=li\_str+"<button class=\"btn btn-default \"  data-toggle = \"modal\"  data-target=\"#del\_building\_modal\" onclick='del\_building(this)'>删除楼号</button>";
25. //console.log(li\_str);
26. $('#add\_building\_li').html("");
27. $('#add\_building\_li').append(li\_str);
28. //拿到多少栋楼
29. });
30. }

以上前端代码的作用是get请求遍历数据库的所有内容然后添加按钮到主界面



1. $.post('init.php',{},function(result)
2. {
3. console.log("初始化"+result)
4. var parsedJson = jQuery.parseJSON(result);
5. var str=''
6. **for** (var i=0;i<parsedJson.length;i++)
7. {
8. **if**(parsedJson[i].building==str\_select\_building)
9. {
10. //这里拼出一个id  先是全局的被选中的建筑号 str\_select\_building 再拼上id first\_1
11. var tmp=str\_select\_building+"\_"+parsedJson[i].ID;
12. str+='<tr id="'+tmp+'tr'+'"><td>'+parsedJson[i].ID+'</td><td>'+parsedJson[i].capacity+'</td><td>'+parsedJson[i].attribute+'</td><td><button class="btn btn-primary btn-xs update" data-toggle = "modal" data-target = "#updateModal" id="'+tmp+'">修改</button><button class="btn btn-danger btn-xs del"  id="'+tmp+'">删除</button></td></tr>'
13. }
14. }
15. //用for循环将json文件里拼接成字符串
16. $(".table").append(str)
17. //将刚才保存的字符串append（在被选元素的结尾（仍然在内部）插入指定内容）到我们的HTML里面
18. });

以上代码是根据选中的楼号去遍历该表中所有会议室的属性，然后添加到页面上



实现以上效果，点击增加，修改，删除后遍可以实现维护数据表的基本操作，选中不同的按钮，通过分成不同的方式（style）去请求admin.php，达到实现不同效果的作用。

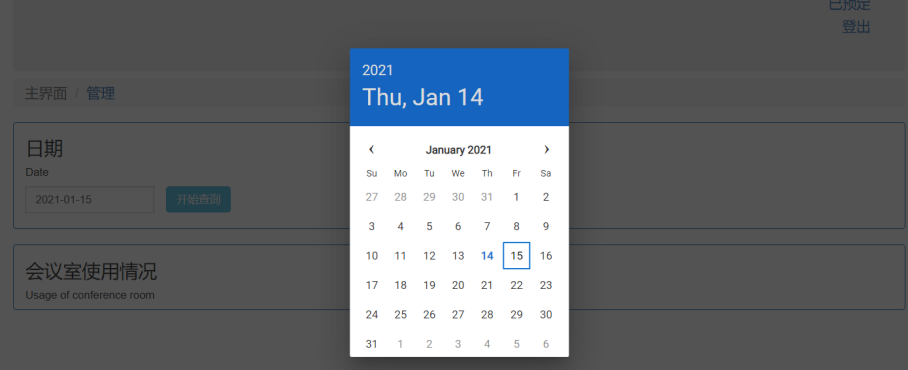
1. **if**($style=="del\_room")
2. {
3. //删除一个房间也就是删除一个行
4. $building\_room=$\_GET['building\_room']; //可能为first\_1
5. $temp= explode('\_',$building\_room);
6. $building=$temp[0];
7. $room\_id=$temp[1];
8. $sql="DELETE FROM $building WHERE ID=$room\_id;";
9. mysqli\_query($con,"set names 'utf8'");
10. mysqli\_query($con,$sql);
11. echo "执行删除";
13. }**elseif**($style=="update\_room")
14. {
15. $building\_room=$\_GET['building\_room'];
16. $temp= explode('\_',$building\_room);
17. $building=$temp[0];
18. $room\_id=$temp[1];
19. // capacity:arr[0],
20. // attributes:arr[1]
21. $capacity=$\_GET['capacity'];
22. $attribute=$\_GET['attribute'];
23. $sql = "UPDATE $building
24. SET capacity='$capacity',attribute='$attribute'
25. WHERE ID='$room\_id';";
26. mysqli\_query($con,"set names 'utf8'");
27. mysqli\_query($con,$sql);
28. echo "执行修改";
30. }**elseif**($style=="add\_room")
31. {
32. //设置一次增加两个房间
33. $building=$\_GET['building'];
34. **for**($a=1;$a<=2;$a++){
35. $ID=$\_GET['id'.$a];
36. $capacity=$\_GET['capacity'.$a];
37. $attribute=$\_GET['attribute'.$a];
38. $sql = "INSERT INTO $building ".
39. "(id,capacity, attribute) ".
40. "VALUES ".
41. "('$ID','$capacity','$attribute');";
42. mysqli\_query($con,"set names 'utf8'");
43. mysqli\_query($con,$sql);
44. }

其中“style”的“del\_room”、“update\_room”和“add\_room”，对应着删除，修改和增加房间的不同操作。

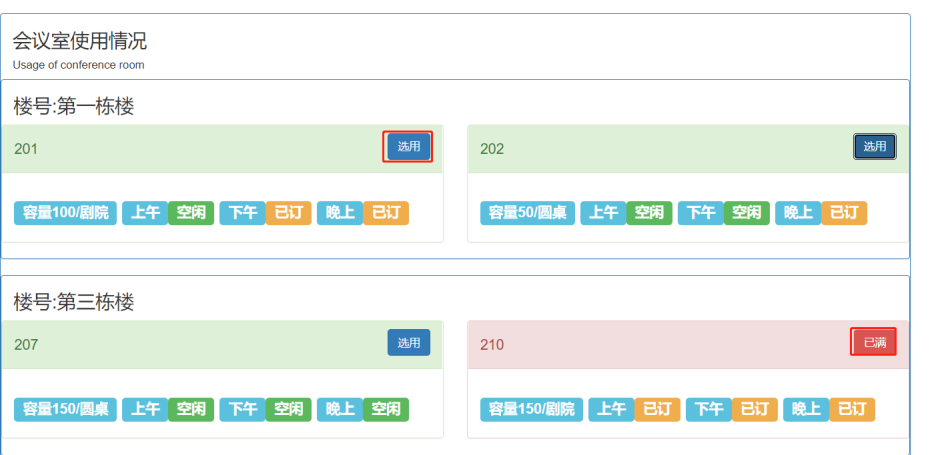
1. **软件运行效果（程序界面及操作说明，如果有数据库则需要说明库表设计）**
2. 登录窗口

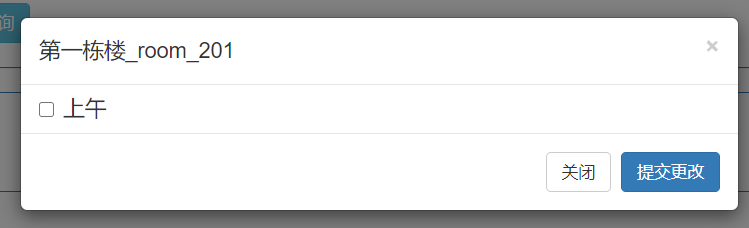


1. 日期选择



1. 根据需求选用时间段

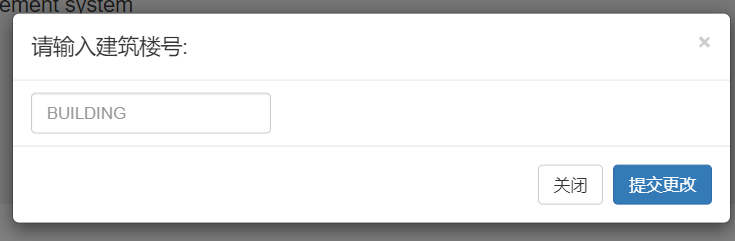


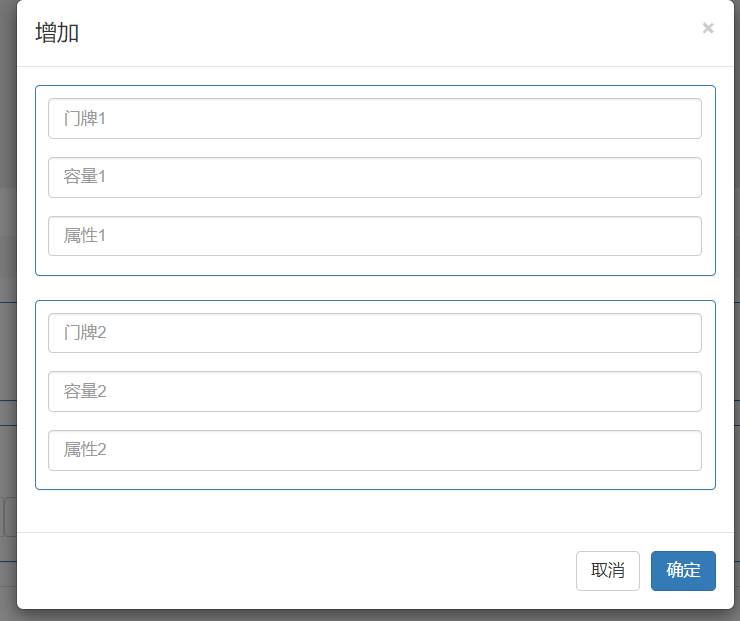


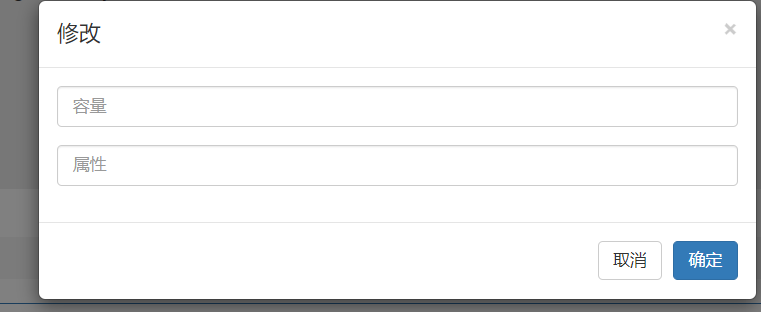


1. 管理界面，实现对数据库会议楼的维护

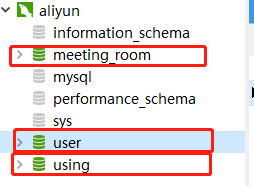




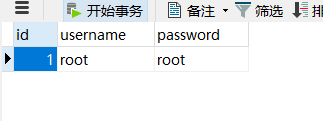


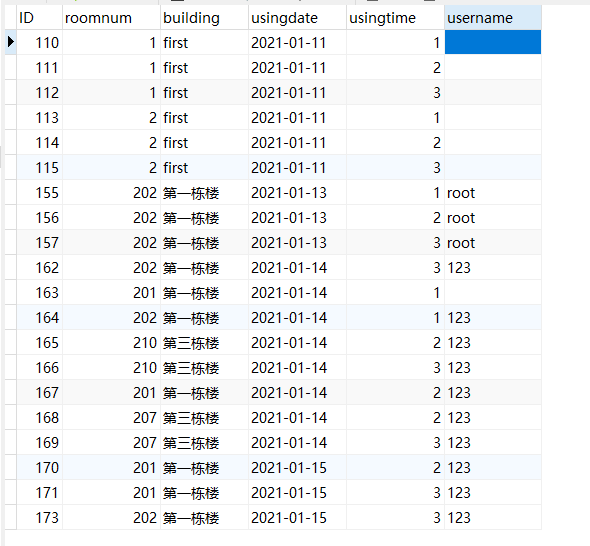


1. mysql数据库页面









1. **实验体会**

本次实验使用前端开发以及后端的流程实现，充分的体会了软件开发的前期功能模块的设计、软件过程的编写以及最后软件的调试测试，学到了如何规范去编写代码以及软件测试相关知识。经过这次学习，使我更加深刻的认识了软件设计，对于后面的学习路线有了更加清楚了解。