

**《Web前端高级技术》**

**课程设计报告**

**设计题目：** 待办清单系统

**院 系：** 计算机科学与工程学院

**专 业：** 　 软件工程

**班　　级：** 软件162

**学 号：** Z09416222

**姓 名：** 徐成

**指导教师：** 周荣

**设计地点：**

**开课时间：** 2017 至 2018 学年第 2 学期

常熟理工学院计算机科学与工程学院 制

**学生姓名**  **成绩**

**指导教师（签名）**

**年 月 日**

**目录**

1. 设计目的和任务.........................................................................................1

2. 开发环境.....................................................................................................2

2.1 硬件环境.............................................................................................2

2.2 软件环境.............................................................................................2

2.3 运行环境.............................................................................................2

3.设计题目.......................................................................................................3

3.1 题目名称.............................................................................................3

3.2 题目详细描述......................................................................................3

3.3 功能要求..............................................................................................3

4. 相关技术及知识点......................................................................................4

4.1用户登录和注销.................................................................................4

4.2添加待办事项.....................................................................................4

4.3标记任务完成.....................................................................................4

4.4新建清单............................................................................................4

4.5隐藏/显示已完成事项........................................................................4

4.6更改显示样式功能..............................................................................4

4.7本地保存清单功能..............................................................................4

4.8查找待办事项功能..............................................................................5

4.9排序功能.............................................................................................5

5. 设计与实现..................................................................................................6

5.1用户登录和注销…….............................................................................6

5.2 新建清单与删除……............................................................................7

5.3 添加代办事项与修改............................................................................9

5.4任务完成与删除....................................................................................10

5.5查找待办事项.....................................................................................11

5.6本地保存清单功能..............................................................................11

5.7 排序功能与优先任务功能..................................................... ............12

6. 总结..............................................................................................................13

7. 参考资料…………………………………….....................................................15

1. 设计目的和任务

1.1项目的目标：

1.综合使用 Node.js，Vue.js，Bootstrap 等技术。

2.灵活使用express技术。

3.掌握数据库相关知识。

1.2项目的任务：

完成代办清单系统，前台用node.js和express和vue.js编程实现。 后台用数据库存储。设计并实现代办清单系统。

具有以下功能：

1. 需要用户登录才能使用待办清单系统，具备用户注销功能。

2. 添加待办事项

3. 标记任务完成

4. 新建清单功能

5. 隐藏/显示已完成事项功能

6. 更改显示样式功能

7. 本地保存清单功能（下次登录网站后还能看到以前的记录）

8. 查找待办事项功能

9. 排序功能

10. 优先任务功能（选做）

11. 拓展功能（选做）

2.开发环境

2.1 硬件环境

电脑型号：戴尔Inspiron 5559 笔记本电脑

操作系统：Windows10 64位

处理器：英特尔Core i5-6200 @ 2.30GHz 双核

内存：4GB

2.2 软件环境

Notepad++文本编译器

Chrome浏览器

Xmapp集成软件包

Navicat Premium数据库管理软件

2.3 运行环境

Node.js

Express

Jquery js框架

3.设计题目

3.1题目名称

代办清单系统

3.2题目详细描述

开发一个简洁的任务管理系统（待办清单系统）来保持一切有序进行。

3.3功能要求

1. 需要用户登录才能使用待办清单系统，具备用户注销功能。

2. 添加待办事项

3. 标记任务完成

4. 新建清单功能

5. 隐藏/显示已完成事项功能

6. 更改显示样式功能

7. 本地保存清单功能（下次登录网站后还能看到以前的记录）

8. 查找待办事项功能

9. 排序功能

10. 优先任务功能（选做）

4.相关技术与知识点

1. 用户登录和注销

要求实现用户登录和注销，所以要用到用户名和密码，要实现对用户名的密码的保存和修改，要设计用户登录界面，点登录后转跳到主界面。主要用到bootstrap的表单样式和按钮样式，还有在数据库存储用户名和密码。还有一些功能如注册用户，检验用户名和密码准确性需要用到js技术。用到了cookie，session会话，保存用户数据和登录信息。

2. 添加待办事项

要求添加代办事项，也是核心功能，首先要对页面布局，写一个div用来存放添加事项页面，页面用表单实现，样式用bootstrap上样式，写函数调用这个功能，为了美观，用jQuery 的 action() 执行对元素的操作，先把页面隐藏，点击后显现，还要实现数据提交和保存功能，把记录同步到数据库，同时要设计修改功能，便于修改数据，这些功能都通过js代码实现。

3. 标记任务完成

要求把已经完成的事项标记并从代办事项中移除，同时存放到已完成清单中，已完成清单相当于一个回收站，可以把事件删除。用到了鼠标点击事件调用函数，彻底删除函数。

4. 新建清单功能

在清单列表中可以新建一个清单，实现方法和新建代表事项差不多，也是用了div隐藏和显现功能，然后函数调用实现功能。

5. 隐藏/显示已完成事项功能

为了实现这个功能，可以建立一个已完成清单，点击这个清单就会显示已经完成的事项，要隐藏这些事项，只要点击其他清单就可以实现隐藏。

6.更改显示样式功能

综合运用bootstrap和css修改样式，使界面更加简洁美观。

7.本地保存清单功能

运用数据库技术，在添加修改删除清单的同时把数据同步写入数据库，等下次登录时再从数据库调用数据，就可以实现本地保存清单功能。

8. 查找待办事项功能

查找待办事项功能只要运用数据库的select语句就能实现，还可以用%实现模糊匹配功能，在查找的同时可以对事项按要求排序。

9.排序功能

在代办事项中设置了紧急度，分为一般，重要，很重要3个等级，分别为1，2，3，然后代办事项根据紧急度排序，很重要排在最前面，然后是重要，一般排在最后面。

5.设计与实现

5.1用户登录和注销

图5.1.1是用户登录界面，写在login.html中，只要输入正确的用户名和密码，就能登录，然后转跳到index.html页面。

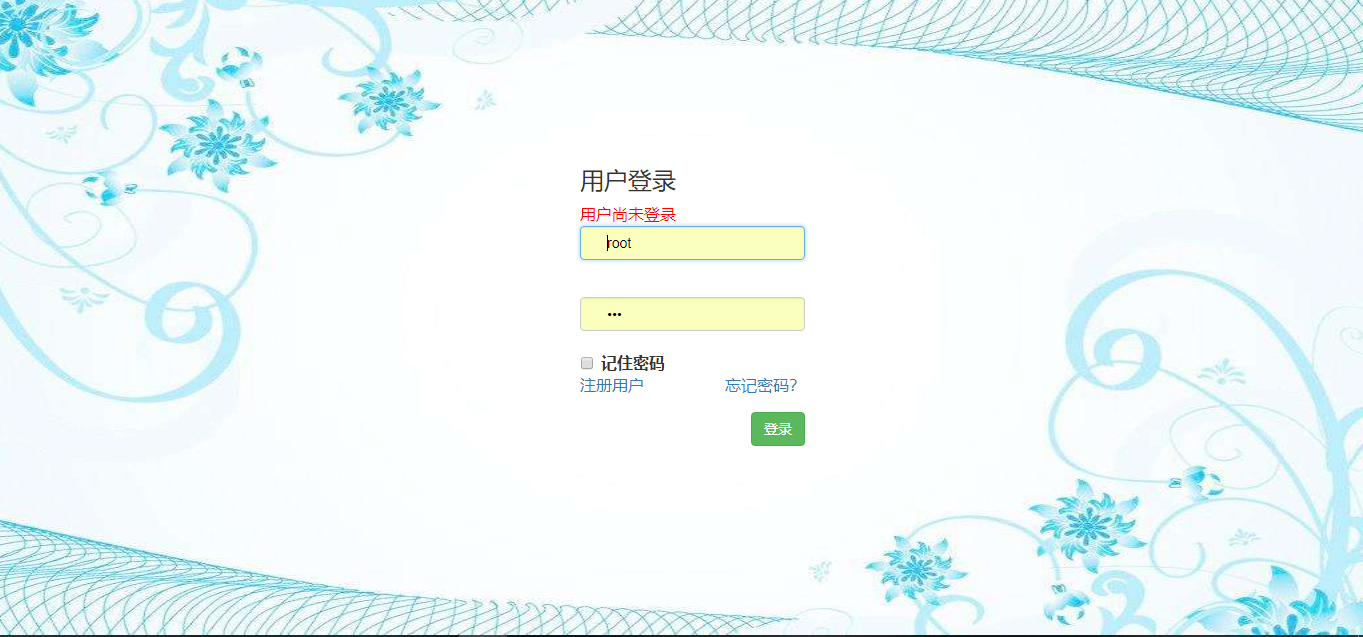


图5.1.1

提示信息

exports.getLogin = (request, response, next) => {

let tips = "";

let tip = request.query.tip;

if(tip == "1") tips = "用户名或密码错误";

else if(tip == "2") tips = "用户尚未登录";

response.render("login", {

"tip": tips

});

}

"=>"代表Arrow Function（箭头函数），相当于匿名函数，可以简化函数定义，就不需要每次打function(request, response, next)了。首先置提示信息tips为空，然后服务器获取tip的值，如果tip等于1，就输出用户名或密码错误，要求重新输入。如果tips值为2，那就是注销后或者未登录情况，就需要进行登录。response.render()函数响应login页面输出tips的信息。

登录实现：

首先应用Formidable中间件，创建Formidable对象from。然后form.parse(request, (err, fields, files)=>方法会转换请求中所包含的表单数据，function(err, fields, files)中包含了所有输入信息，如用户名，密码等。如果出现错误，则会输出表单解析失败，然后执行next(),回调form.parse(request, (err, fields, files)=>功能，继续解析表单数据。

解析成功后，把提取到的username，password赋给要检验的用户名，密码。编写sql语句，查询数据库里login表中的usernamr和password，然后调用dao.js中的connection.query(sql, function(err, result)方法，调用sql语句查询用户名和密码。如果调用过程中出现错误，则输出错误信息，然后用next()方法调用下一个中间件，没有错误就继续执行下一步操作。如果没有从数据库中查询到结果，也就是用户名密码为空，则向请求的客户端发送响应头，重定向到登录页面：

response.writeHead(302, { "Location": "http://127.0.0.1:3000/login?tip=1" });。

否则就将从数据库查询到的用户名提交给服务器由session会话保存，然后向请求的客客户端发送响应头，转到用户界面：

response.writeHead(302, {"Location": "http://127.0.0.1:3000/index?list=2"});，

最后结束响应，登录完成。



图5.1.2

图5.1.2是登录成功的页面，会显示用户的头像，旁边是注销按钮，点击则会回到登录界面。

exports.getLogout = (request, response, next) => {

request.session.userName = null;

response.writeHead(302, {

"Location": "http://127.0.0.1:3000/login?tip=2"

});

response.end();

}

注销功能是将session会话中保存的uesername置为空，然后回到login页面中tip=2的界面，要求重新登录或者更换账号登录。

2.新建和删除清单



图5.2.2

图5.2.2是新建清单功能，它用到了隐藏和显示div功能，首先在屏幕中央设计一个名为”cover”的div，设置好样式，然后用no\_addList()功能将其隐藏，这个function的代码如下：

function hide\_addList(){

$("#addList").hide();

}

function hide\_cover(){

$("#cover").hide();

}

function no\_addList(){

hide\_addList();

hide\_cover();

}

如果要新建清单，单击按钮，就会显示这个div，就可以实现功能了，用到了addlist()功能，代码如下：

function show\_addList(){

$("#addList").show();

}

function show\_cover(){

$("#cover").show();

}

function addList(){

show\_addList();

show\_cover();

}

这些都是jQuery 的 action() 执行对元素的操作，然后实现对元素的隐藏和显现。

同时在提交的同时调用check\_list()功能，保证清单名不为空，代码如下：

function check\_list(){

if(document.getElementById("addList\_input").value == "")return false;

return true;

}

新建清单具体实现在router.js中的AddList()函数中实现，首先检查session中的username是否为空，如果为空则跳转到登录页面，如果不为空，继续执行。然后应用formidable表单中间件，创建Formidable对象from，然后解析表单内容，如果解析出错，就输出错误信息，然后回调，如果解析成功，就把提取出的列表名赋给listname，编写插入列表名的sql语句，调用dao.js 中的add()方法，把列表名插入到数据库中，如果插入出现错误，就显示错误信息，如果没有出现错误，就向请求的客客户端发送响应头，转到list\_id为2的界面，也就是to-do界面。

删除清单：点击删除按钮

function removeList(id,list){

location.href = "/removeList?id="+id+"&list="+list;

}

这个功能由router.js中的RemoveList()方法实现，如果检查用户名没有问题，就用jquery获取列表传给list，然后获取列表id，编写sql语句删除取得id值的列表，调用dao.js中的remove功能，就可以执行sql语句删除列表，删除列表后，仍是跳转到lis\_id 为这个值的界面，但是内容为空。

5.3 添加代办事项与修改



图5.3.1

如图5.3.1，这个功能和上面的新建清单有点类似，也是用到了隐藏/显示div功能。首先调用no\_addthings()功能，把”covers”这个div隐藏：

function no\_addthings(){

hide\_addthings();

hide\_covers();

}

function hide\_addthings(){

$("#addthings").hide();

}

function hide\_covers(){

$("#covers").hide();

}

当点击添加代办事项时，显示”covers”

function show\_addthings(){

$("#addthings").show();

}

function show\_covers(){

$("#covers").show();

}

function addthings(){

show\_addthings();

show\_covers();

}

然后输入信息，点击提交，调用router.js中的AddThing()功能，首先验证用户名，出错就回到登录页面重新登录。验证正确后，用formidable中间件，创建Formidable对象from，把list\_id置为2。

如果表单中内容不为空，就解析表单，提取输入信息。解析成功后，把这些信息分别赋给id,title,urange,day等参数，如果输入的数字是个位数，就在前面加个0，使其更美观。最后把日期信息连接成year-month-day的形式。

接着编写sql语句，插入信息，用？代替，然后把插入参数赋给params。如果list\_id不为空，编写sql语句，更新信息，把从表单获取的信息更新到list中，接着调用dao.js中的update功能实现数据更新，如果出现错误，显示错误信息，然后回调，如果没有错误，就完成添加事项操作，跳转到当前页面。

如果list\_id为空，就直接执行添加新建事项功能，然后也是转跳到当前页面。

修改数据功能：

点击修改按钮，会出现修改界面，可以修改数据，修改页面和添加页面是同一个页面

调用index.js中的updateThing功能，把屏幕上的信息提取到表单中显示出来，接着调用router.js中addthings()方法提交修改信息。

取消功能:

取消修改，把所有提取到的信息置为空，防止下一次新建代办事项出现原来的数据。这个功能在index.js中的qiuxiao()方法中实现。

图5.3.2是修改数据：



图5.3.2

5.4 任务完成与删除

点击完成任务按钮，会显示任务完成，事项会移动到已完成清单，点击删除任务，可以删除已完成事项。

任务完成：

调用router.js中的DeleteThing()功能，首先验证用户名是否正确，不正确就退出重新登陆，验证正确后获取要完成事项的id和当前列表的id，编写sql语句把该事项的isdelete置为1，表示事项已经完成，然后isdelete为1的事项都在已完成列表中，接着调用dao.js中update功能，实现更新操作把isdelete变为1，最后重定向页面到当前页面，任务完成操作结束。

彻底删除：调用router.js中的RemoveThing功能，获取已完成事项的id值，编写sql语句，从数据库中删除id为该值的事项，应用dao.js中的remove功能删除事项，最后重定向页面到当前页面，也就是id为3的页面。

图5.4.1为效果图



图5.4.1

5.5查找待办事项

效果图：



图5.5.1

实现：

调用router.js中Search()功能，因为搜索页面用的formidable中间件，可以新建一个表单对象from，然后解析表单，提取出搜索关键词赋值给search参数，编写sql语句进行搜索操作，用的是模糊匹配，搜索含有该关键词的所有事项，然后输出到搜索结果中，这个搜索只能搜索未完成的事项，标记已完成的事项无法搜索，因为搜索出的事项可以进行操作，如完成，修改等，但是已完成的事项无法操作，只能删除，这两个无法同时操作，所以只搜索未完成的 。

搜索成功后用sql语句查询lists表，用response.render功能渲染视图模板把搜索成功的页面的id改为-1，并输出搜素结果。

5.6本地保存清单功能

本项目的数据都保存在数据库中，根据dao.js中的功能更新和删除数据，再更新和删除的功能中调用sql语句对数据库中的数据进行操作，在数据库中保存数据，所以可以实现本地保存清单功能。

5.7 排序功能和优先任务功能

在代办事项中设置了紧急度，分为一般，重要，很重要3个等级，分别为1，2，3，然后代办事项根据紧急度排序，很重要排在最前面，具体实现在getindex()方法中。

首先如果session会话中保存的用户名为空，则跳转到登录页面重新登录，然后设置id为2，从to-do清单开始，用parseInt(request.query.list)函数获取列表的id值，定义sql语句从数据库查询新创建的清单列表，就是id为3以后的列表的id，因为在前端已经给前3个列表赋了id值也就是list\_flag，调用dao.js中的query函数进行查询操作，left\_result代表查询出的清单列表。

当id=1，也就是我的一天页面，首先获取当前日期，也就是年，月，日，然后把这些赋给year，month，day参数，最后把这些参数用“-”连接起来赋给now\_date作为当前日期，定义sql语句从代办清单里找到与当天日期相同的清单添加到我的一天，按照紧急度降序排列，这样就可以把事情紧迫的放到前面，先去完成。

当id为2时，也就是to-do页面，同样定义sql语句把isdelete为0的事项筛选出来，也就是未完成事项，按紧急度降序排序，时间升序排列，这就可以实现优先排序功能，把快到截至日期的，紧迫的放到前面，督促用户完成。

当id为3时，是已完成页面，把isdelete为1的事项筛选出来，也就是已完成事项，按紧急度降序排序，时间升序排列。

对于新建的列表中的事项，也是通过这种方法排序。只要把id值获取就能实现操作。

最后调用dao.js的query方法实现sql语句，实现排序优先任务功能，用response.render功能渲染视图模板在index的右边返回排序结果，左边返回清单信息。

6.总结：

在做项目的时候学到了很多，也遇到了不少问题，最大的收获是学会了 express的使用。

在前端部分，Div的布局很重要，布局要简单，清晰，不然到后面div变多后都不知道哪个是哪个，有可能你改了这个div的样式，结果呈现出来的是另一个div，这就很麻烦。还有div的名字也很有讲究，不要乱起名字，一开始我的名字都是随便打的，到后来都不知道这个到底是什么东西了，要起一些你看的懂的名字，后面修改时就轻松多了。

第二是样式的设计，bootstrap就很好用，就可以省略很多css代码，还很美观，你要做的就是在这基础上对样式作很小的改动，如移动或者改变大小颜色什么的。不过也有可能出现问题，很多地方都用到了同一个按钮，它的class是相同的，在css文件中改了其中一个样式，另一个也会随之改变，这不是我希望看到的，所以我们可以用行内样式直接修改样式，或者给按钮另起名字，然后分开修改。

还有就是功能的调用，很容易出错，有时候一个功能一开始还能用，当你把这个功能调到别的地方去了就不能用了。比如说在设计添加代办清单功能时，代办清单都写在表单里，这个在之前的实验中就能实现了，而新建清单就在代办清单下面，添加好了就可以直接生成在下面，不过考虑到长度有限，有时候备注很长，一个格子写不下，就需要另起一行，这就影响了美观。于是就想设计成注册页面那样的，点击新建就跳出一个页面，在这个页面上填写信息，但如果要转跳到另一个网页这就很麻烦，就想到了隐藏div这个功能。首先先设计个div，稍微大点，显示在屏幕中央，然后再调用功能，把这个div隐藏掉，当点击添加代表清单时，调用功能，把那个div显示出来，这样就很方便了。这个功能只要用jQuery 的 action() 执行对元素的操作就能实现隐藏和显现。

不过这样设计后就出现了另一个问题，就是修改功能，修改功能和添加功能共用一个表单，当点击修改时突然没反应，这让我很纳闷，一开始可以修改的啊。当我无意点到添加代办清单按钮时，竟然出现了修改页面，于是我重新去看代码，发现修改功能要调用事项的信息，刚好这个功能在添加中实现，于是在修改功能中加入添加功能，就可以直接实现修改了。如果修改完成后点提交，那不会出现问题，但是如果修改到一半用户不想修改了，就把这个界面点掉了，而当你再次去新建代表清单时就会出现问题，它不是显示空白信息，而是会显示上次未完成修改的信息。为了解决这个问题，添加了一个取消按钮，点击后可以清空所有信息，这样在下次重新添加时就不会出现问题了。

新建清单功能的方法和新建代办事项差不多 ，也是div隐藏，左边是一整个导航栏，用了bootstrap的样式，前面小按钮也是用bootstrap完成的，比插入图片方便多了，可以随时设置样式。删除清单按钮也是直接用bootstrap上提供的按钮，点击后可以删除清单。不过这部分的样式很复杂，删除按钮无法完美的融入到导航栏中，后来把他们放到同一个li中才解决了问题，不然按钮一直悬在右下角很难受。

然后用户登录和注销功能的实现就用到页面重定向，在login页面，如果用户名密码都输入正确，就会跳转到index页面，这就是登录，而点击退出按钮，就会返回到login页面，这就是用户注销。同时还有用户名密码的检测功能，保证用户名密码的正确性，还有注册用户功能和记住密码功能。

至于后端的设计，主要用的是node.js和express，express需要我们自己去学习，它的官网上给了我们很多的基础知识和入门操作，比如说可以用express应用生成器快速创建一个应用的框架。还有一些关于基本路由的介绍，通过路由可以实现我们编写的函数。还有利用 Express 托管静态文件，如image，css，js等，

将静态资源文件所在的目录作为参数传递给express.static中间件就可以提供静态资源文件的访问了，这样在调用js，css时就不用写很长的路径了。在express中用到了很多次中间件，也就是next变量，它可以执行任何代码，修改请求和响应对象，终结请求-响应循环，调用下一个中间件。所以在函数的错误处理中很多用到了中间件，防止请求被挂起。还有response.render方法可以渲染html，然后将渲染得到的HTML文档发送给客户端。

在很多函数中用到了函数response.writeHead(statusCode, [reasonPhrase], [headers])，这个方法是用来写响应头部信息的，这个应该在发送body主体之前调用，statusCode是状态码，我们经常看到的404就是这样的状态码，表示告诉客户端没有找到资源，而2xx这一类型的状态码，代表请求已成功被服务器接收、理解、并接受，3xx代表重定向，也是这个项目中用到的，可以在结束响应后跳转到其他页面，4xx表示请求错误，5xx表示服务器错误。特别是3xx一类的状态码如300，301，302等，我们用的基本都是302状态码，因为这是临时的，客户端应当继续向原有地址发送以后的请求，这样就不会永久移动到新位置。

我们很多地方用到了formidable中间件，这是一个关于表单的中间件，可以处理表单和上传文件，这里我们用来处理表单，获取表单上的输入信息，然后把它们传入参数，最后用这些参数编写sql语句实现对数据库中的数据的操作。

还有一个就是jquery js框架，有些功能就是通过里面的功能实现的，比如说在index页面隐藏一个如何显示它，就用到了$(this).hide() 和$(this).show() ，只要这两句代码就可以实现功能。

参考资料：

1. 菜鸟教程 [www.runoob.com](http://www.runoob.com)
2. Express官网 [www.expressjs.com.cn](http://www.expressjs.com.cn)
3. 《深入浅出node.js》 作者:朴灵 出版社:人民邮电出版社
4. 《Node与express开发》(美) Ethan Brown著 出版社:人民邮电出版社