

Vue+Springboot+MyBatis 技术应用解析

陈倩怡, 何军

(中国人民银行长沙中心支行, 长沙 410005)

摘要: 采用流行的框架 Vue+Springboot+Mybatis 进行业务开发, 对关键模块进行了具体解析, 给出了开发方法和应当关注的方面, 对 B/S 应用开发具有参考价值。

关键词: Vue; springboot; mybatis; axios; 切面技术

DOI:10.16184/j.cnki.comprg.2020.01.005

1 概述

Vue 结合 Springboot 及 MyBatis 是当下 B/S 开发模式的热门技术, Vue 重点在视图层的开发应用上, vue-cli 是官方发布的脚手架, 可快速搭建前端框架, 其运用到的关键技术包括单页面开发、Vue 路由、Vue 状态管理, axios 请求和响应技术。Springboot 应用于后端开发, 其是 Spring 开发技术的升级, 引入注解, 简化了 Web 框架的配置和开发流程。Mybatis 是对象映射框架, 内置 jdbc, 只关注 SQL 本身。其前端的开发过程如下: Vue 单页面 template 部分结合 iview 控件布置页面布局, script 部分定义 Javascript 脚本, style 部份定义设置 class 样式。在 script 中通过 axios 的 post 方法与后端服务器进行数据交互。后端开发过程如下: 通过 Mybatis 自动生成工具 Mybatis-generator-core 生成表对应的映射文件、model 类及 DAO 类, 分别置于应用目录的 mapping、entity 及 Dao, 对代码作微调, Dao 继承 basedao 完成公用的增、删、改、查功能, 映射文件将自动生成的增、删、改、查的配置 id 改名, 否则运行会报错。接下来编写 service 类自动装配 dao 类, 调用 dao 方法。最后编写 controller 类与读取前端传入参数, 调用 sevice 方法, 将运算结果反馈给前端。

2 技术应用解析

2.1 操作员管理模块 (Vue 前端)

实现的功能包括: 界面分为左右两部份, 左面树型结构, 列出操作员姓名, 右边为 TAB 页第一页列出操作员的一般信息, 包括用户代码、用户姓名、口令、工作状态、帐户状态、是否业务管理员。第二至第六页分别对操作单位角色、应用系统角色、操作部门角色、操作菜单角色、人员类别角色进行授权, 分别使用 iview 控件穿梭框实现。操作角色编码作为唯一键用作穿梭框的 Key 值, 为管理不同 tab 页中角色的状态, 使用 Vue

的 Store 进行状态管理, 在穿梭框按钮的 onchange 事件中应用 this.store.commit ("updateyyxts", newTargetKeys), 调用 store 中的 mutations 更新每个 tab 页中用户对操作权限赋予的角色。在保存按钮中通过 vm.yyxts=vm.\$store.state.app.yyxts;vm.yyxts.forEach (item => { vm.user-role.push (item)}) 读取用户赋予的操作角色, 通过 axios 发送到后台写入数据库中。读取 store 的状态值后台保存时 Vue 存在一个 BUG, 第一个值需要通过 vm.user-role=vm.\$store.state.app.unit; vm.\$store.commit ("updateunits", vm.\$store.state.app.unit); 读取后马上要回写一次, 否则 userrole 后续读取的状态值在连续点击保存的情况下, 每次不重新刷新而是累加上次读取的状态值, 导致读取的状态值失真。组合权限按钮显示所有角色所拥有的组合操作权限, 以应用系统角色为例, 左边树型控件显示所有的应用系统, 右边为表格控件, 显示该用户所具有的可操作的应用系统, 树型控件子节点根据用户可操作和不可操作的应用系统分别渲染成勾和叉的自定义图标, 自定义图标从阿里巴巴图标库下载, 将下载的文件夹拷贝到目录 static 下, 在欲引入的 Vue 文件 style 中通过 @import './././static/iconfont/iconfont.css' 引入, 在 script 程序段通过 class: 'iconfont iconseleted' 设置自定义图标。

2.2 菜单角色权限 MyBatis 原生态 SQL 实现 (springboot 后端)

操作员管理模块的显示用户可操作的菜单需从后台获取权限数据, 菜单分三级菜单, 一级菜单、二级菜单, 三级菜单分别存储在不同的表中, 反映操作员权限

作者简介: 陈倩怡 (1967-), 女, 本科, 高级工程师, 研究方向: 计算机软件; 何军 (1969-), 男, 硕士, 高级工程师, 研究方向: 计算机系统软件工程。

SOFTWARE DEVELOPMENT & APPLICATION

的表存在另一个表中，通过一个 SQL 语句一次性获取权限数据，SQL 语句结构比较复杂，有左连接及排序；这种情况下，常规的 MyBatis 面向对象的方法无法实现，可以改用原生态 SQL 完成功能。第一步在 `gr_cd1gn-mapper.xml` 中书写 SQL 语法：

```
<select id="selectMenusByyhdm" resultType="java.util.Map" parameterType="java.lang.String">
    ${_parameter}
</select>
```

返回值为 Map 集合；第二步在 dao 层中书写接口函数 `List<Map<String, Object>> selectMenusByyhdm(String sql)`；第三步在 service 层中书写调用函数：

```
@Override
public List<Map<String, Object>> selectMenusByyhdm(HashMap<String, Object> map)
{
    String sql = "(select distinct t.cd1xh,t.cd2xh,t.cd3xh,t1.cd1mc,t2.cd2mc,t3.cd3mc from gr_role_qx t, gr_cd1gn t1, gr_cd2gn t2, gr_cd3gn t3, gr_role_set t4, gr_czy_role t5 where t5.yhdm='"+map.get("yhdm")+"' and t5.role_no=t4.role_no and t4.role_type='"+map.get("role_type")+"' and t4.child_role_no=t.child_role_no ";
    sql = sql + "and t.cd1xh=t1.cd1xh and t.cd1xh=t2.cd1xh (+) and t.cd2xh=t2.cd2xh (+) and t.cd1xh=t3.cd1xh (+) and t.cd2xh=t3.cd2xh (+) and t.cd3xh=t3.cd3xh(+) order by cd1xh,cd2xh,cd3xh";
    List<Map<String, Object>> result = gr_cd1gnMapper.selectMenusByyhdm(sql);
    return result;
};
```

第四步在 controller 层中写处理 http 连接的函数，处理与 vue 前端的连接。`public MenuList getMenu(@RequestBody Map<String, Object> reqMap, HttpServletRequest request)`

其基本思路是：获取 Vue 前端传入的 map 参数，将 map 参数分解，调用 service 层服务函数得到权限数据，其中有两次查询，一次得所用菜单，一次得用户可操作菜单，将结果封装成对象返回给 Vue 前端展示。

2.3 文件上传模块 (Vue 前端)

上传模块界面用 `iview` 控件 `upload` 读取上传文件名，`<upload :before-upload="handleUpload1">`
`<Button icon="ios-cloud-upload-outline">浏览</Button>`
`</upload>`
 用 `formData` 读取上传文件数据，

```
handleUpload1 (file) {
```

```
    this.file1 = file;
    this.filepath1=file.name;
    return false;
},
```

```
    upload(){
        var formData = new FormData();
        formData.append('file', this.file1);
        通过 axios 的 post 与后台通信，将文件数据传送到后台
        var config = 'multipart/form-data';
        var urlStr="/upload/oneFile";
        Util.ajax.post(urlStr,formData,config)
        .then(function (response)。
```

2.4 文件接收模块 (Springboot 后端)

```
public String oneFile(@RequestParam(value = "file",required = false) MultipartFile files, HttpServletRequest req)throws IOException {
    String uploadPath="D:/photos";
    //构建上传的文件夹
    File dir = new File(uploadPath);
    if(! dir.exists()){
        dir.mkdir();
    }
    MultipartHttpServletRequest multipartHttpServletRequest=(MultipartHttpServletRequest)req
    List<MultipartFile> list=multipartHttpServletRequest.getFiles("file");
    for(MultipartFile multifile :list ) {
        String path = dir + "/" + multifile.getOriginalFilename();
        File newFile = new File(path);
        multifile.transferTo(newFile);
    }
```

使用 bean 设置文件上传大小，

```
MultipartConfigFactory factory1 = new MultipartConfigFactory();
//单个文件最大
Factory1.setMaxFileSize("200MB");
// 设置总上传数据大小
Factory1.setMaxRequestSize("200MB");
return factory1.createMultipartConfig();
```

将该 bean 放置到启动类中即生效。

2.5 切面技术实现日志插入 (Springboot 后端)

以切面技术实现通过注解扩展方法代码，在方法上
 (下转第 28 页)



设中,使图书馆的环境、服务、管理、资源更加智能化,可以通过实体图书馆和虚拟网络平台两种途径进行管理服务,争取为线上线下相结合的渠道构建整体化服务体系。通过物联网技术实现对用户和图书馆的智慧化服务和管理,改造传统意义上的图书馆。智慧图书馆的建设内容包括环境智慧化、服务智慧化、管理智慧化和资源智慧化4个方面。

4.3 物联网校园安防

智慧校园安防应以保障学生和教职员工的人身安全为重点,是智慧校园的重要功能之一。它采用光纤、无线等传输网络,运用计算机技术、图像技术、物联网技术等,实时、形象、真实地对校园进行视频监控和电子巡查。校园安防系统一般包括视频监控、入侵报警、门禁管理、车辆管理和安防综合监管平台5部分。智能人脸识别门禁系统,正在各中职院校和中小学开始运用。使学校门口真正实现“智慧化”管理。学校门口安装智能闸机,学生老师“刷脸”出入校园,彻底解决学校的安全管理和考勤管理,与以往的监控相比更安全,是学校智慧校园领域精心打造的又一力作。

4.4 物联网校务管理

智慧校园的校务管理功能主要体现在用信息化工具即云计算、大数据等新技术来优化学校资源配置,提高学校行政和组织效率,对教育教学进行预测和规划,促进管理方法的科学化和管理模式的智慧化,进而形成新

的管理模型,提高学校管理工作的水平。建立智慧校园校务管理平台,是教育现代化发展的必然趋势,是智慧校园建设的重要任务。

4.5 物联网校园文化

校园文化是一种特殊的社会文化现象,是智慧校园的又一个重要功能。它是以建设有中国特色的社会主义文化为根基,是学校在长期教育教学实践中积累形成的,老师和学生共同遵循的价值观念和行为准则。智慧校园文化一般由硬件设备及软件系统构成,主要包括校园内网络、数字广播、网络电视、交互智能平板显示屏、数字标牌、校园信息发布系统主题互动展示软件及主题电子签名软件等。学校架构内外部网络以及服务器提供外部和内部的网络服务;专业场景应用软件与终端硬件结合,实现互动展示。

5 结语

结合国内中职院校教学条件以及学习者特征,提出符合我国中职院校的物联网实施的方案,打造智慧校园,为广大师生提供更安全、方便、快捷的教学环境,实现资源的整合和共享,真正实现中职院校将物联网应用于智慧校园。

参考文献

- [1] 刘修文. 物联网技术应用——智慧校园. 机械工业出版社.
- [2] 黄玉兰. 物联网概论. 人民邮电出版社.

(上接第15页)

添加注解 @MyLog1 (value=" 明细查询测试",modelName="所有明细查询"),方法执行完后,执行添加的切面方法,切入点执行 saveSyslog,

```
MyLog myLog1=method.getAnnotation(MyLog1.class);
if(myLog1!=null){
    String value=myLog1.value();
    syslog.setRemark(value);
    syslog.setOpName(myLog1.modelname());
}
```

将注解添加的 value 和 modelname 参数引入 syslog 类对象中, syslog 读取用户代码、主机名、IP 地址、操作日期、操作时间形成一条日志信息记录,插入日志记

录表中。

3 结语

Vue、Springboot、MyBatis 作为项目开发的三驾马车,是 B/S 开发的黄金组合,易于上手,易于开发、调试和维护。

参考文献

- [1] 梁灏. Vue.js 实战. 清华大学出版社, 2017.
- [2] 王福强. SpringBoot 揭秘: 快速构建微服务体系. 机械工业出版社, 2016.
- [3] 刘增辉. MyBatis 从入门到精通. 电子工业出版社, 2017.