

基于 Vue 的通用后台管理系统设计与实现

姜全坤

(广东理工学院, 广东 肇庆 526100)

摘 要: 后台管理系统作为 Web 应用不可或缺的一部分, 使 Web 应用可以方便、快捷地管理内容。从零开始开发一个后台管理系统往往需要投入大量的人力和物力。为此, 文章使用 Vue 框架实现了一个通用的后台管理系统, 该系统具有后台管理的一般功能, 开发人员可以在该系统的基础上进行二次开发, 显著提高了开发效率。

关键词: 后台管理系统; Vue 框架; Mock 请求拦截

中图分类号: TP311

文献标识码: A

文章编号: 2096-4706 (2022) 08-0033-04

Design and Implementation of a Universal Background Management System Based on Vue

JIANG Quankun

(Guangdong Technology College, Zhaoqing 526100, China)

Abstract: As an indispensable part of Web application, background management system enables Web application to manage content conveniently and quickly. Developing a background management system from scratch often requires a lot of human and material resources. Therefore, this paper uses Vue framework to implement a general background management system. This system has the general functions of background management. Developers can carry out secondary development on the basis of this system, which significantly improves the development efficiency.

Keywords: background management system; Vue framework; Mock request interception

0 引言

在互联网如此发达的今天, Web 应用得到了很好的发展。人们在日常使用电脑的过程中访问各种各样的网页, 在网上浏览前沿知识都离不开互联网和 Web 应用的发展。每一个 Web 应用通常都有前台和后台两套系统, 前台系统用于向用户展示信息 (也就是用户在浏览器上看到的网页内容), 后台系统可供网站管理人员管理网站内容 (例如, 博客后台系统^[1]可以方便作者编辑和删除博客)。然而, 后台管理系统的开发往往是耗时费力的, 开发人员需要进行需求分析、系统设计、系统实现、系统测试等。不同后台管理系统之间有很多共同点, 但是一个后台管理系统的开发又不能直接照搬其他后台管理系统的模式。因此, 开发一个规范、扩展性好的通用后台管理系统是十分有必要的。

1 相关技术介绍

1.1 Vue.js

Vue.js^[2] 是基于 MVVM 模式开发的, 主要用于构建用户界面, 可以显著提高开发人员的开发效率。它具有模板语法、双向数据绑定、计算属性、监听器、条件渲染、列表渲染、事件处理、表单绑定、组件化应用等诸多特性, 使开发人员可以将工作重点集中在系统设计上面, 从而实现快速开发。

1.2 Element UI

Element UI^[3] 是一款前端组件库, 它为开发人员提供了一种高效便捷的构建前端界面的方式。Element 基于 Vue 开发, 支持 Vue 的许多新特性, 是前端开发人员必须掌握的一个组件库。Element 具有丰富的组件库和图标库, 每个组件都有自己对应的属性和事件, 开发人员只需根据组件的要求提供相应的属性和函数即可使用组件。

1.3 Mock.js

Mock.js^[4] 是一个生成随机数据和链接 Ajax 请求的库, 它在提倡前后端分离的今天非常实用。Mock.js 可以模拟后台数据并拦截前端发起的请求, 还可以根据设定返回给前端相应的数据, 这为前端工程师提供了极大的便利。

2 系统需求分析

通用后台管理系统要想做到通用, 就必须实现绝大多数后台管理系统所具备的功能。本文系统提供了后台管理基本布局, 该布局主要由左右两部分组成, 左侧是导航菜单, 它提供了首页、通用一级菜单和通用二级菜单, 开发人员可根据需求添加所需菜单; 右侧又分为上下两部分: 右侧上面部分为头部信息, 该区域左侧有一个菜单显示与隐藏按钮, 它可以控制左侧菜单的隐藏与显示, 中间是导航面包屑, 它可以告知用户当前所在的位置, 右侧是用户头像显示区域, 用户点击用户头像可展示下拉菜单。右侧下面部分为页面标签和页面内容区域, 页面标签位置会显示用户已打开的页面标签, 用户可以关闭某些已打开的标签。页面主要内容区域会

根据路由显示不同的页面。系统框架图如图1所示。

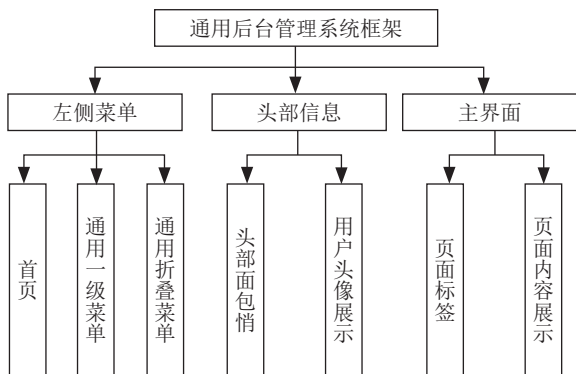


图1 系统总体框架图

3 系统总体设计

3.1 系统主要配置

系统主要配置主要集中在 main.js 文件中, 在这里需要引入 element-ui 和与其对应的样式文件, 需要引入 mock.js 来拦截 Ajax 请求, 需要引入 vue-router 路由控制页面的跳转, 还需要引入 vuex 管理相应数据以供不同组件之间通信。其核心代码为:

```
import router from '@router'
import store from '@store'
import '@api/mock/index'
import ElementUI from 'element-ui';
import 'element-ui/lib/theme-chalk/index.css';
Vue.use(ElementUI, {size: 'small'});
new Vue({router, store, render: h => h(App),
  created() {
    store.commit('addMenu', router)
  }
}).$mount('#app')
```

3.2 左侧菜单设计

左侧菜单提供后台管理系统的主要功能展示和导航, 因此设计时应考虑到开发者常用的系统名称与 LOGO 展示、首页、一级菜单和二级菜单等功能, 效果如图2所示。



图2 左侧菜单效果图

左侧菜单核心代码为:

```
<template>
```

```
<el-menu default-active="1-4-1" class="el-menu-vertical-demo"
  :collapse="isCollapse" background-color="#545c64" text-color="#fff"
  active-text-color="#ffd04b" router>
  <h3>
    <span v-show="!isCollapse"> 后台管理系统 </span></h3>
  <el-menu-item @click="clickMenu(item)"
    v-for="item in noChildrenMenu" :index="item.path" :key="item.label">
    <i :class=" 'el-icon-' + item.icon "></i>
    <span slot="title">{{ item.label }}</span>
  </el-menu-item>
  <el-submenu v-for="item in haveChildrenMenu" :index="item.path"
    :key="item.path"><template slot="title">
    <i :class=" 'el-icon-' + item.icon "></i>
    <span slot="title"> 导航一 </span></template>
    <el-menu-item @click="clickMenu(item1)"
      v-for="item1 in item.children" :index="item1.path" :key="item1.label">
    <i :class=" 'el-icon-' + item1.icon "></i>
    <span slot="title">{{ item1.label }}</span>
  </el-menu-item></el-submenu></el-menu>
</template>
```

3.3 顶部展示区域设计

该区域提供一个左侧菜单伸缩按钮, 它可以控制左侧菜单的显示与隐藏。中间部分是面包屑功能, 它可以快速便捷地显示用户所在位置, 点击不同的面包屑按钮可以跳转至相应的页面。右侧是用户头像显示区域, 该区域可以显示用户头像, 用户点击头像会展开下拉菜单, 方便用户修改个人信息或退出, 顶部展示区域效果如图3所示。



(a) 顶部面包屑



(b) 头像下拉菜单

图3 顶部展示区域效果图

其核心代码为:

```
<template><header><div class="l-content">
  <el-button size="mini" icon="el-icon-menu"
    style="margin-right: 10px"></el-button>
```

```

<el-breadcrumb separator= "/" >
<el-breadcrumb-item v-for= "item in tags" :key= "item.
path"
      :to= "{ path: item.path }" class= "bdColor" >
      {{ item.label }}</el-breadcrumb-item></el-breadcrumb></
div>
<div class= "r-content" >
<el-dropdown trigger= "click" @command= "logout" ><span>
<img class= "user" src= "~@/assets/avatar.jpg"
alt= "" /></span>
<el-dropdown-menu slot= "dropdown" >
<el-dropdown-item command= "a" > 个 人 中 心 </el-
dropdown-item>
<el-dropdown-item command= "b" > 退 出 </el-
dropdown-item>
</el-dropdown-menu></el-dropdown></div></header>
</template>

```

3.4 主界面设计

主界面会根据路由展示不同的页面，这里以用户管理界面为例，该界面实现了用户的增、删、改、查功能。增加用户和编辑用户使用的是公共的表单组件，这样可以更加高效地利用组件并提高开发效率，其效果图如图 4 所示。

姓名	年龄	性别	出生日期	地址	操作
姜芳	39	女	2001-11-16	宁夏回族自治区	编辑 删除
张超	41	女	1992-02-06	内蒙古自治区	编辑 删除
杜娟	66	男	2016-11-27	浙江省	编辑 删除
郑平	36	女	2003-04-23	上海	编辑 删除
孔涛	19	女	1996-11-11	浙江省	编辑 删除

图 4 用户管理界面效果图

其核心代码为：

```

<template><div class= "common-table" >
<el-table :data= "tableData" stripe style= "width:
100%" >
<el-table-column v-for= "item in tableLabel"
:key= "item.prop"
      :label= "item.label" :width= "item.width ? item.
width : 90"
      show-overflow-tooltip style= "height:90%" >
<template slot-scope= "scope" >
<span style= "margin-left: 10px" >{{ scope.row[item.
prop] }}</span>
</template></el-table-column>
<el-table-column label= "操作" >
<template slot-scope= "scope" >
<el-button @click= "handleEdit(scope.row)"
size= "mini" > 编辑
</el-button><el-button @click= "handleDelete(scope.
row)"

```

```

size= "mini" type= "danger" > 删除 </el-button></template>
</el-table-column></el-table>
<el-pagination background layout= "prev, pager, next"
:total= "config.total"
      :current-page.sync= "config.page" @current-
change= "changePage"
      :page-size= "10" class= "pager" ></el-pagination></
div>
</template>

```

4 系统测试

在项目文件夹中使用 yarn serve 命令启动项目。在浏览器地址栏中输入：http://localhost:8080/login 进入登录界面，用户输入正确的用户名和密码登录系统，如果输入的密码错误会有错误提示，其效果如图 5 所示。



图 5 用户登录测试

不同用户由于拥有的权限不同（使用动态路由^[5]控制），登录系统后左侧菜单会显示不同的内容，这样可以限制无权限用户的操作，图 6 显示不同权限用户菜单。



(a) 管理员菜单



(b) 普通用户菜单

图 6 不同权限用户菜单

如用户打开多个页面，页面标签栏会显示已打开的页面标签，用户可以通过点击标签上面的 × 号关闭特定的标签，其效果如图 7 所示。



图 7 已打开页面标签

5 结 论

综上所述，本文叙述了一个通用后台管理系统的设计与实现，该系统基于 Vue.js 等相关技术构建，基于前后端分离的思想，实现了包括用户登录、用户管理、多（下转 38 页）

级二类赛、行业赛已成功举办三届,分别为2017年“川仪杯”首届全国仪器仪表制造工职业技能竞赛、2019年“中控杯”第二届全国仪器仪表制造工职业技能竞赛、2020年第三届“中控杯”全国仪器仪表制造工职业技能竞赛,提高了行业影响力。

“川仪杯”全国首届仪器仪表制造工职业技能竞赛决赛于2017年3月24日至26日在重庆举办,由中国机械工业联合会、中国就业培训技术指导中心、中国机冶建材工会全国工作委员会联合主办,中国仪器仪表学会、中国四联仪器仪表集团有限公司、机械工业职业技能鉴定指导中心承办。2019年全国智能装备制造及管理职业技能竞赛——“中控杯”第二届全国仪器仪表制造工赛项总决赛于2019年12月5至8日在河南化工技师学院举办,有来自全国30多家仪器仪表企业及院校,44个参赛队,111名选手同台竞技。由中国机械工业联合会、中国就业培训技术指导中心、中国机械冶金建材工会全国委员会主办,中国仪器仪表学会、河南化工技师学院承办的2020年第十八届全国机械行业职业技能竞赛——“中控杯”第三届全国仪器仪表制造工赛项决赛于2020年11月27至29日在河南化工技师学院举办,有来自全国仪器仪表企业及院校的122名选手同台竞技。

由于前期良好的工作基础,加上学会积极、努力申报,《仪器仪表制造工国家职业技能标准》也正式颁布,“仪器仪表制造工大赛”于2021年成功升级为国家级一类赛,意义重大,标志着仪器仪表行业,在中国最高职业技能大赛舞台上有一席之地。在2021年第四届智能制造应用技术技能大赛——仪器仪表制造工(仪器仪表与智能传感应用技术)赛项中,全国30个省、市人社厅、局组织开展省级选拔赛,最终全国100多支参赛队200多人参加决赛,是行业内非常好的技能大练兵活动,也是一项高级科普活动。

2022年,学会将开展人社部国家级二类赛“2022年全国行业职业技能竞赛——第五届全国仪器仪表行业职业技能竞赛”,通过技能大赛,让更多职业院校了解仪器仪表在整

个国民经济发展以及智能制造中的地位和作用,也带动整个仪器仪表行业职业教育产业发展,包括学科专业建设、人才队伍建设、职业教育培训、教材课程体系开发、评价体系建立,以及行业转型升级等,都具有积极的推动作用。

3 结论

行业职业技能标准的制定需符合行业未来发展,体现实用性及先进性,而只有本行业的科技社团组织有能力、有担当去承担该项任务。对于仪器仪表制造工过去几十年均没有完整的国家职业技能标准的情况,中国仪器仪表学会充分发挥科技社团行业引领示范作用,积极主动作为,制订《仪器仪表制造工国家职业技能标准》,该标准的制订和颁布,为高职学校、中专院校制订技能人才培养方案提供依据,为培养高技能人才、大国工匠做出贡献,为实施制造强国战略和推动高质量发展提供有力人才支撑。

参考文献:

- [1] 王雪,张莉.中国仪器仪表工程科技2035发展趋势研究[J].中国工程科学,2017,19(1):103-107.
 - [2] 中国工程院.制造强国战略研究[M].北京:电子工业出版社,2015.
 - [3] 中华人民共和国工业和信息化部.加快推进传感器及智能化仪器仪表产业发展行动计划[R].北京:中华人民共和国工业和信息化部,2013.
 - [4] 中华人民共和国工业和信息化部.“十二五”装备制造业发展和结构调整的思路、目标、重点及对策研究[R].北京:中华人民共和国工业和信息化部,2010.
 - [5] 中华人民共和国人力资源和社会保障部.国家职业技能标准——仪器仪表制造工[EB/OL].(2020-11-13)[2022-03-25].<http://chinajob.mohrss.gov.cn/zcms/contentcore/resource/download?ID=88152>.
 - [6] 国家职业分类大典修订工作委员会.中华人民共和国职业分类大典:2015年版[M].北京:中国劳动社会保障出版社,2015.
- 作者简介:张迎春(1982.11—),女,汉族,四川大竹人,工程师,本科,研究方向:职业能力发展与管理。

(上接35页)级菜单、页面导航、权限控制、数据可视化等功能。该系统具有较好的适用性和扩展性,开发人员可以在此系统的基础上快速开发出适合自己公司场景的后台管理系统,该系统可以显著提高开发人员的开发效率。

虽然该系统提供了后台管理系统的通用功能,但是该系统并没有提供更多的组件来满足不同公司的特殊需求。另外,开发人员需要掌握该系统的开发思想和思路才能在此系统的基础上进行开发,这对部分开发人员来说需要投入一定的学习成本。因此,在后续的系统开发中需要不断改进和完善该系统。

参考文献:

- [1] 姜全坤,郭卓恒,闫娇娇.基于SpringBoot+AdminLTE3

的博客管理系统设计与实现[J].信息与电脑(理论版),2021,33(24):124-127.

- [2] 赵龙,王风硕.基于Vue的图书销售系统设计与实现[J].电脑知识与技术,2021,17(36):81-82+85.

- [3] 曾晓钰,唐莹,温丰蔚,等.一种基于ElementUI的表格查询组件开发方案[J].现代工业经济和信息化,2021,11(12):50-51+56.

- [4] 秦子实.企业内网环境中基于Mock.js的前端应用独立调试方法[J].电脑知识与技术,2018,14(17):54-56.

- [5] 师明,曾丹.基于Vue.js和Spring Boot的校招日记系统[J].工业控制计算机,2020,33(1):95-97.

作者简介:姜全坤(1992—),男,汉族,河南新乡人,助教,硕士,研究方向:机器学习。