

实验 11：完成管理系统前端开发

一、实验目的

- 1、掌握 View UI Plus 组件库在 Vue 3 中的使用方法
- 2、熟练使用组合式 API 开发前端应用
- 3、理解现代化管理系统界面设计原则
- 4、培养组件化开发和状态管理能力

二、实验学时

2 学时

三、实验类型

综合性

四、实验需求

1、硬件

每人配备计算机 1 台，建议优先使用个人计算机开展实验。

2、软件

安装 Visual Studio Code，以及 Edge 浏览器。

3、网络

本地主机能够访问互联网和实验中心网络。

4、工具

无。

五、实验任务

- 1、使用Vite创建View UI Plus项目
- 2、实现中医药数据管理界面布局
- 3、开发药材、方剂、功效等核心功能模块
- 4、实现数据的增删改查操作

六、实验内容及步骤

1、项目初始化与配置

步骤 1：创建Vite项目

Bash

```
1 # 使用官方模板创建项目
2 npm create vue@latest tcm-management
3 cd tcm-management
4 npm install
5
6 # 安装View UI Plus
7 npm install view-ui-plus @vicons/ionicons5
```

步骤 2：配置View UI Plus

JavaScript

```
1 // main.js
2 import { createApp } from 'vue'
3 import App from './App.vue'
4 import router from './router'
5 import ViewUIPlus from 'view-ui-plus'
6 import 'view-ui-plus/dist/styles/viewuiplus.css'
7
8 const app = createApp(App)
9 app.use(router)
10 app.use(ViewUIPlus)
11 app.mount('#app')
```

步骤 3：项目结构规划

Plain Text

```
1 src/
2 |— components/          # 公共组件
3 |   |— Layout/         # 布局组件
4 |   |— DataTable/      # 数据表格组件
5 |   |— SearchBar/      # 搜索组件
6 |— views/              # 页面组件
7 |   |— HerbManagement.vue  # 药材管理
8 |   |— PrescriptionManagement.vue # 方剂管理
9 |   |— EfficacyManagement.vue  # 功效管理
10 |— utils/              # 工具函数
11 |   |— api.js          # API接口
12 |— assets/             # 静态资源
```

2、主布局组件实现

步骤 1：创建主布局组件

```
1 <!-- src/components/Layout/MainLayout.vue -->
2 <template>
3   <Layout class="layout">
4     <!-- 顶部导航 -->
5     <Header class="header">
6       <div class="header-content">
7         <div class="logo">
8           <Icon type="ios-medical" size="24" />
9           <span class="logo-text">中医药数据管理系统</span>
10        </div>
11        <div class="user-info">
12          <Avatar icon="ios-person" />
13          <span class="username">管理员</span>
14          <Dropdown @on-click="handleUserAction">
15            <Icon type="ios-arrow-down" />
16            <template #list>
17              <DropdownMenu>
18                <DropdownItem name="profile">个人中心</DropdownItem>
19                <DropdownItem name="logout" divided>退出登录</DropdownItem>
20              </DropdownMenu>
21            </template>
22          </Dropdown>
23        </div>
24      </div>
25    </Header>
26
27    <Layout>
28      <!-- 侧边栏 -->
29      <Sider collapsible :width="220" class="sider">
30        <Menu
31          :active-name="activeMenu"
32          theme="dark"
33          width="auto"
34          @on-select="handleMenuSelect"
35        >
36          <MenuItem name="herbs" to="/herbs">
37            <Icon type="ios-leaf" />
38            <span>中药材管理</span>
39          </MenuItem>
40          <MenuItem name="prescriptions" to="/prescriptions">
41            <Icon type="ios-medkit" />
```

```
42         <span>中药方剂</span>
43     </MenuItem>
44     <MenuItem name="efficacy" to="/efficacy">
45         <Icon type="ios-flask" />
46         <span>功效分类</span>
47     </MenuItem>
48     <MenuItem name="patients" to="/patients">
49         <Icon type="ios-people" />
50         <span>患者管理</span>
51     </MenuItem>
52 </Menu>
53 </Sider>
54
55 <!-- 主内容区 -->
56 <Content class="content">
57     <router-view />
58 </Content>
59 </Layout>
60 </Layout>
61 </template>
62
63 <script setup>
64 import { ref, computed } from 'vue'
65 import { useRoute, useRouter } from 'vue-router'
66 import { Message } from 'view-ui-plus'
67
68 const route = useRoute()
69 const router = useRouter()
70
71 // 计算当前激活的菜单
72 const activeMenu = computed(() => {
73     return route.name || 'herbs'
74 })
75
76 // 处理菜单选择
77 const handleMenuSelect = (name) => {
78     console.log('选择菜单:', name)
79 }
80
81 // 处理用户操作
82 const handleUserAction = (action) => {
83     if (action === 'logout') {
84         Message.success('退出登录成功')
85         router.push('/login')
```

```
86   }
87 }
88 </script>
89
90 <style scoped>
91 .layout {
92   height: 100vh;
93 }
94
95 .header {
96   background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
97   padding: 0;
98 }
99
100 .header-content {
101   display: flex;
102   justify-content: space-between;
103   align-items: center;
104   height: 64px;
105   padding: 0 24px;
106 }
107
108 .logo {
109   display: flex;
110   align-items: center;
111   color: white;
112   font-size: 18px;
113   font-weight: bold;
114 }
115
116 .logo-text {
117   margin-left: 12px;
118 }
119
120 .user-info {
121   display: flex;
122   align-items: center;
123   color: white;
124   gap: 8px;
125 }
126
127 .sider {
128   background: #001529;
129 }
```

```
130
131 .content {
132   padding: 24px;
133   background: #f5f7f9;
134   overflow: auto;
135 }
136 </style>
```

3、药材管理模块实现

步骤 1：药材管理页面

```
1 <!-- src/views/HerbManagement.vue -->
2 <template>
3   <Card>
4     <template #title>
5       <div class="page-header">
6         <h2><Icon type="ios-leaf" /> 中药材管理</h2>
7         <Button type="primary" icon="md-add" @click="showModal('add')">
8           新增药材
9         </Button>
10      </div>
11    </template>
12
13    <!-- 搜索区域 -->
14    <div class="search-area">
15      <Row :gutter="16">
16        <Col :span="6">
17          <Input
18            v-model="searchParams.name"
19            placeholder="药材名称"
20            clearable
21            @on-clear="handleSearch"
22          >
23            <template #prefix>
24              <Icon type="ios-search" />
25            </template>
26          </Input>
27        </Col>
28        <Col :span="4">
29          <Select v-model="searchParams.category" placeholder="药材分类"
30            clearable>
31            <Option value="根茎类">根茎类</Option>
32            <Option value="果实类">果实类</Option>
33            <Option value="花叶类">花叶类</Option>
34            <Option value="全草类">全草类</Option>
35          </Select>
36        </Col>
37        <Col :span="4">
38          <Select v-model="searchParams.nature" placeholder="药性" clear
39            able>
40            <Option value="寒">寒</Option>
41            <Option value="热">热</Option>
42            <Option value="温">温</Option>
```

```
41         <Option value="凉">凉</Option>
42         <Option value="平">平</Option>
43     </Select>
44 </Col>
45 <Col :span="4">
46     <Button type="primary" @click="handleSearch">搜索</Button>
47     <Button style="margin-left: 8px" @click="handleReset">重置</Bu
48     </Col>
49 </Row>
50 </div>
51
52 <!-- 数据表格 -->
53 <Table
54     :columns="columns"
55     :data="herbList"
56     :loading="loading"
57     border
58     stripe
59 >
60     <template #nature="{ row }">
61         <Tag :color="getNatureColor(row.nature)">{{ row.nature }}</Tag>
62     </template>
63
64     <template #taste="{ row }">
65         <span v-for="taste in row.taste" :key="taste" class="taste-ta
66             {{ taste }}
67         </span>
68     </template>
69
70     <template #status="{ row }">
71         <Switch
72             v-model="row.status"
73             true-value="1"
74             false-value="0"
75             @on-change="(val) => updateStatus(row.id, val)"
76         >
77             <span slot="open">启用</span>
78             <span slot="close">禁用</span>
79         </Switch>
80     </template>
81
82     <template #action="{ row }">
```

```
83     <ButtonGroup>
84         <Button type="primary" size="small" @click="showModal('edit', row)">
85             编辑
86         </Button>
87         <Button type="info" size="small" @click="viewDetail(row)">
88             详情
89         </Button>
90         <Button type="error" size="small" @click="handleDelete(row)">
91             删除
92         </Button>
93     </ButtonGroup>
94 </template>
95 </Table>
96
97 <!-- 分页 -->
98 <div class="page-wrapper">
99     <Page
100         :total="total"
101         :current="pagination.current"
102         :page-size="pagination.pageSize"
103         @on-change="handlePageChange"
104         show-total
105         show-elevator
106     />
107 </div>
108 </Card>
109
110 <!-- 新增/编辑模态框 -->
111 <Modal
112     v-model="modalVisible"
113     :title="modalTitle"
114     width="600"
115     @on-ok="handleSubmit"
116     @on-cancel="handleCancel"
117 >
118     <Form
119         ref="formRef"
120         :model="formData"
121         :rules="formRules"
122         :label-width="100"
123     >
124         <FormItem label="药材名称" prop="name">
125             <Input v-model="formData.name" placeholder="请输入药材名称" />
```

```
126     </FormItem>
127
128     <FormItem label="拼音名称" prop="pinyin">
129       <Input v-model="formData.pinyin" placeholder="请输入拼音名称" />
130     </FormItem>
131
132     <FormItem label="药材分类" prop="category">
133       <Select v-model="formData.category" placeholder="请选择分类">
134         <Option value="根茎类">根茎类</Option>
135         <Option value="果实类">果实类</Option>
136         <Option value="花叶类">花叶类</Option>
137         <Option value="全草类">全草类</Option>
138         <Option value="树皮类">树皮类</Option>
139         <Option value="矿物类">矿物类</Option>
140       </Select>
141     </FormItem>
142
143     <FormItem label="性味归经" prop="nature">
144       <Row :gutter="8">
145         <Col :span="8">
146           <Select v-model="formData.nature" placeholder="药性">
147             <Option value="寒">寒</Option>
148             <Option value="热">热</Option>
149             <Option value="温">温</Option>
150             <Option value="凉">凉</Option>
151             <Option value="平">平</Option>
152           </Select>
153         </Col>
154         <Col :span="16">
155           <Select
156             v-model="formData.taste"
157             multiple
158             placeholder="药味（可多选） "
159           >
160             <Option value="辛">辛</Option>
161             <Option value="甘">甘</Option>
162             <Option value="酸">酸</Option>
163             <Option value="苦">苦</Option>
164             <Option value="咸">咸</Option>
165           </Select>
166         </Col>
167       </Row>
168     </FormItem>
169
```

```

170     <FormItem label="功效主治" prop="efficacy">
171       <Input
172         v-model="formData.efficacy"
173         type="textarea"
174         :rows="3"
175         placeholder="请输入功效主治"
176       />
177     </FormItem>
178
179     <FormItem label="用法用量" prop="usage">
180       <Input v-model="formData.usage" placeholder="请输入用法用量" />
181     </FormItem>
182
183     <FormItem label="注意事项" prop="precautions">
184       <Input
185         v-model="formData.precautions"
186         type="textarea"
187         :rows="2"
188         placeholder="请输入注意事项"
189       />
190     </FormItem>
191   </Form>
192
193   <template #footer>
194     <Button @click="handleCancel">取消</Button>
195     <Button type="primary" :loading="submitLoading" @click="handleSub
196       mit">
197       确定
198     </Button>
199   </template>
200 </Modal>
201 <!-- 详情抽屉 -->
202 <Drawer
203   v-model="detailVisible"
204   title="药材详情"
205   width="600"
206   >
207   <Descriptions v-if="currentHerb" bordered>
208     <DescriptionsItem label="药材名称">{{ currentHerb.name }}</Descript
209     ionsItem>
210     <DescriptionsItem label="拼音名称">{{ currentHerb.pinyin }}</Descri
211     ptionsItem>
212     <DescriptionsItem label="药材分类">{{ currentHerb.category }}</Desc

```

```

        rptionsItem>
211         <DescriptionsItem label="性味">
212             <Tag :color="getNatureColor(currentHerb.nature)">{{ currentHer
b.nature }}</Tag>
213             <span v-for="taste in currentHerb.taste" :key="taste" class="ta
ste-tag">
214                 {{ taste }}
215             </span>
216         </DescriptionsItem>
217         <DescriptionsItem label="功效主治" span="3">
218             {{ currentHerb.efficacy }}
219         </DescriptionsItem>
220         <DescriptionsItem label="用法用量">{{ currentHerb.usage }}</Descrip
tionsItem>
221         <DescriptionsItem label="注意事项" span="2">
222             {{ currentHerb.precautions }}
223         </DescriptionsItem>
224     </Descriptions>
225 </Drawer>
226 </template>
227
228 <script setup>
229 import { ref, reactive, onMounted, computed } from 'vue'
230 import { Message, Modal } from 'view-ui-plus'
231
232 // 响应式数据
233 const herbList = ref([])
234 const loading = ref(false)
235 const modalVisible = ref(false)
236 const detailVisible = ref(false)
237 const submitLoading = ref(false)
238 const modalType = ref('add')
239 const currentHerb = ref(null)
240 const total = ref(0)
241 const formRef = ref(null)
242
243 // 分页配置
244 const pagination = reactive({
245     current: 1,
246     pageSize: 10
247 })
248
249 // 搜索参数
250 const searchParams = reactive({

```

```
251   name: '',
252   category: '',
253   nature: ''
254 })
255
256 // 表单数据
257 const formData = reactive({
258   id: '',
259   name: '',
260   pinyin: '',
261   category: '',
262   nature: '',
263   taste: [],
264   efficacy: '',
265   usage: '',
266   precautions: '',
267   status: '1'
268 })
269
270 // 表单验证规则
271 const formRules = {
272   name: [
273     { required: true, message: '药材名称不能为空', trigger: 'blur' }
274   ],
275   category: [
276     { required: true, message: '请选择药材分类', trigger: 'change' }
277   ],
278   nature: [
279     { required: true, message: '请选择药性', trigger: 'change' }
280   ]
281 }
282
283 // 表格列配置
284 const columns = [
285   { type: 'index', width: 60, align: 'center' },
286   { title: '药材名称', key: 'name', minWidth: 120 },
287   { title: '拼音名称', key: 'pinyin', width: 120 },
288   { title: '分类', key: 'category', width: 100 },
289   { title: '药性', slot: 'nature', width: 80, align: 'center' },
290   { title: '药味', slot: 'taste', minWidth: 150 },
291   { title: '功效', key: 'efficacy', minWidth: 200, ellipsis: true },
292   { title: '状态', slot: 'status', width: 100, align: 'center' },
293   { title: '操作', slot: 'action', width: 200, align: 'center' }
294 ]
```

```
295
296 // 计算属性
297 const modalTitle = computed(() => {
298   return modalType.value === 'add' ? '新增药材' : '编辑药材'
299 })
300
301 // 方法定义
302 const loadHerbs = async () => {
303   loading.value = true
304   try {
305     // 模拟API调用
306     await new Promise(resolve => setTimeout(resolve, 500))
307
308     // 模拟数据
309     herbList.value = [
310       {
311         id: 1,
312         name: '人参',
313         pinyin: 'Renshen',
314         category: '根茎类',
315         nature: '温',
316         taste: ['甘', '微苦'],
317         efficacy: '大补元气，复脉固脱，补脾益肺，生津养血，安神益智',
318         usage: '3-9g，另煎兑服',
319         precautions: '不宜与藜芦同用',
320         status: '1'
321       },
322       {
323         id: 2,
324         name: '黄芪',
325         pinyin: 'Huangqi',
326         category: '根茎类',
327         nature: '温',
328         taste: ['甘'],
329         efficacy: '补气升阳，固表止汗，利水消肿，生津养血',
330         usage: '9-30g',
331         precautions: '表实邪盛者不宜用',
332         status: '1'
333       }
334     ]
335     total.value = herbList.value.length
336   } catch (error) {
337     Message.error('加载数据失败')
338   } finally {
```

```
339     loading.value = false
340   }
341 }
342
343 const showModal = (type, row = null) => {
344   modalType.value = type
345   modalVisible.value = true
346
347   if (type === 'edit' && row) {
348     Object.assign(formData, { ...row })
349   } else {
350     // 重置表单
351     Object.assign(formData, {
352       id: '',
353       name: '',
354       pinyin: '',
355       category: '',
356       nature: '',
357       taste: [],
358       efficacy: '',
359       usage: '',
360       precautions: '',
361       status: '1'
362     })
363   }
364 }
365
366 const handleSubmit = async () => {
367   const valid = await formRef.value.validate()
368   if (!valid) return
369
370   submitLoading.value = true
371   try {
372     // 模拟API调用
373     await new Promise(resolve => setTimeout(resolve, 1000))
374
375     if (modalType.value === 'add') {
376       const newHerb = {
377         id: Date.now(),
378         ...formData
379       }
380       herbList.value.unshift(newHerb)
381       Message.success('新增成功')
382     } else {
```

```
383     const index = herbList.value.findIndex(item => item.id === formDa
    ta.id)
384     if (index !== -1) {
385         herbList.value.splice(index, 1, { ...formData })
386         Message.success('修改成功')
387     }
388 }
389
390 modalVisible.value = false
391 } catch (error) {
392     Message.error('操作失败')
393 } finally {
394     submitLoading.value = false
395 }
396 }
397
398 const handleCancel = () => {
399     modalVisible.value = false
400     formRef.value.resetFields()
401 }
402
403 const viewDetail = (row) => {
404     currentHerb.value = row
405     detailVisible.value = true
406 }
407
408 const handleDelete = (row) => {
409     Modal.confirm({
410         title: '确认删除',
411         content: `确定要删除药材"${row.name}"吗? `,
412         onOk: async () => {
413             try {
414                 const index = herbList.value.findIndex(item => item.id === row.
                    id)
415                 if (index !== -1) {
416                     herbList.value.splice(index, 1)
417                     Message.success('删除成功')
418                 }
419             } catch (error) {
420                 Message.error('删除失败')
421             }
422         }
423     })
424 }
```

```
425
426 const updateStatus = async (id, status) => {
427   try {
428     const herb = herbList.value.find(item => item.id === id)
429     if (herb) {
430       herb.status = status
431       Message.success(status === '1' ? '启用成功' : '禁用成功')
432     }
433   } catch (error) {
434     Message.error('操作失败')
435   }
436 }
437
438 const handleSearch = () => {
439   pagination.current = 1
440   loadHerbs()
441 }
442
443 const handleReset = () => {
444   Object.assign(searchParams, {
445     name: '',
446     category: '',
447     nature: ''
448   })
449   pagination.current = 1
450   loadHerbs()
451 }
452
453 const handlePageChange = (page) => {
454   pagination.current = page
455   loadHerbs()
456 }
457
458 const getNatureColor = (nature) => {
459   const colors = {
460     '寒': 'blue',
461     '热': 'red',
462     '温': 'orange',
463     '凉': 'green',
464     '平': 'default'
465   }
466   return colors[nature] || 'default'
467 }
468
```

```
469 // 生命周期
470 onMounted(() => {
471   loadHerbs()
472 })
473 </script>
474
475 <style scoped>
476 .page-header {
477   display: flex;
478   justify-content: space-between;
479   align-items: center;
480   margin-bottom: 0;
481 }
482
483 .search-area {
484   margin-bottom: 16px;
485   padding: 16px;
486   background: #f8f8f9;
487   border-radius: 4px;
488 }
489
490 .taste-tag {
491   display: inline-block;
492   padding: 2px 6px;
493   margin: 2px;
494   background: #f0f0f0;
495   border-radius: 3px;
496   font-size: 12px;
497 }
498
499 .page-wrapper {
500   text-align: right;
501   margin-top: 16px;
502 }
503 </style>
```

4、路由配置和主应用

步骤 1：配置路由

JavaScript

```
1 // src/router/index.js
2 import { createRouter, createWebHistory } from 'vue-router'
3
4 const routes = [
5   {
6     path: '/',
7     redirect: '/herbs'
8   },
9   {
10    path: '/herbs',
11    name: 'HerbManagement',
12    component: () => import('@/views/HerbManagement.vue')
13  },
14  {
15    path: '/prescriptions',
16    name: 'PrescriptionManagement',
17    component: () => import('@/views/PrescriptionManagement.vue')
18  },
19  {
20    path: '/efficacy',
21    name: 'EfficacyManagement',
22    component: () => import('@/views/EfficacyManagement.vue')
23  }
24 ]
25
26 const router = createRouter({
27   history: createWebHistory(),
28   routes
29 })
30
31 export default router
```

步骤 2: 主应用文件

```
1 <!-- src/App.vue -->
2 <template>
3   <MainLayout />
4 </template>
5
6 <script setup>
7 import MainLayout from '@components/Layout/MainLayout.vue'
8 </script>
9
10 <style>
11 * {
12   margin: 0;
13   padding: 0;
14   box-sizing: border-box;
15 }
16
17 body {
18   font-family: 'Helvetica Neue', Helvetica, 'PingFang SC', 'Hiragino Sans GB', 'Microsoft YaHei', sans-serif;
19   background-color: #f5f7f9;
20 }
21
22 #app {
23   min-height: 100vh;
24 }
25 </style>
```

七、实验考核

本实验考核采用【实验随堂查】方式开展。

每个实验完成后，在实验课上通过现场演示的方式向实验指导教师进行汇报，并完成现场问答交流。

每个实验考核满分 100 分，其中实验成果汇报 60 分，现场提问交流 40 分。

实验考核流程：

- (1) 学生演示汇报实验内容的完成情况，实验指导老师现场打分。
- (2) 指导老师结合实验内容进行提问，每位学生提问 2-3 个问题，根据回答的情况现场打分。
- (3) 实验考核结束后，进行公布成绩。