# 可视化任务指导书

[可视化任务指导书 1](#_Toc114727985)

[一、 实验效果 2](#_Toc1940412536)

[二、 实验目的 2](#_Toc826829210)

[三、 实验原理 3](#_Toc151852733)

[1，组件封装 3](#_Toc978310895)

[2，实现复杂表格配置 3](#_Toc1336410833)

[3，实现任务类型的切换 4](#_Toc525406258)

[4，实现任务详情的展示与收起 5](#_Toc50221742)

[四、 实验环境 6](#_Toc115744523)

[1． 开发语言 6](#_Toc1845497526)

[2． 开发环境 6](#_Toc1170605861)

[3． 调试环境 6](#_Toc1275015660)

[4． Vue-cli配置 6](#_Toc1596367854)

[5． Nginx服务器 6](#_Toc1641320207)

[五、 实验内容 6](#_Toc1241273334)

[1． 页面布局 6](#_Toc1424777580)

[2． 组件调用 6](#_Toc1794123010)

[3． 接口调用 6](#_Toc1007541543)

[4． 实训任务工作量对标公司工作量 7](#_Toc842156606)

[六、 功能思路分析 7](#_Toc61359665)

[七、 关键技术点解决方案 7](#_Toc479739095)

[八、 实验步骤 7](#_Toc1321358827)

## 实验效果



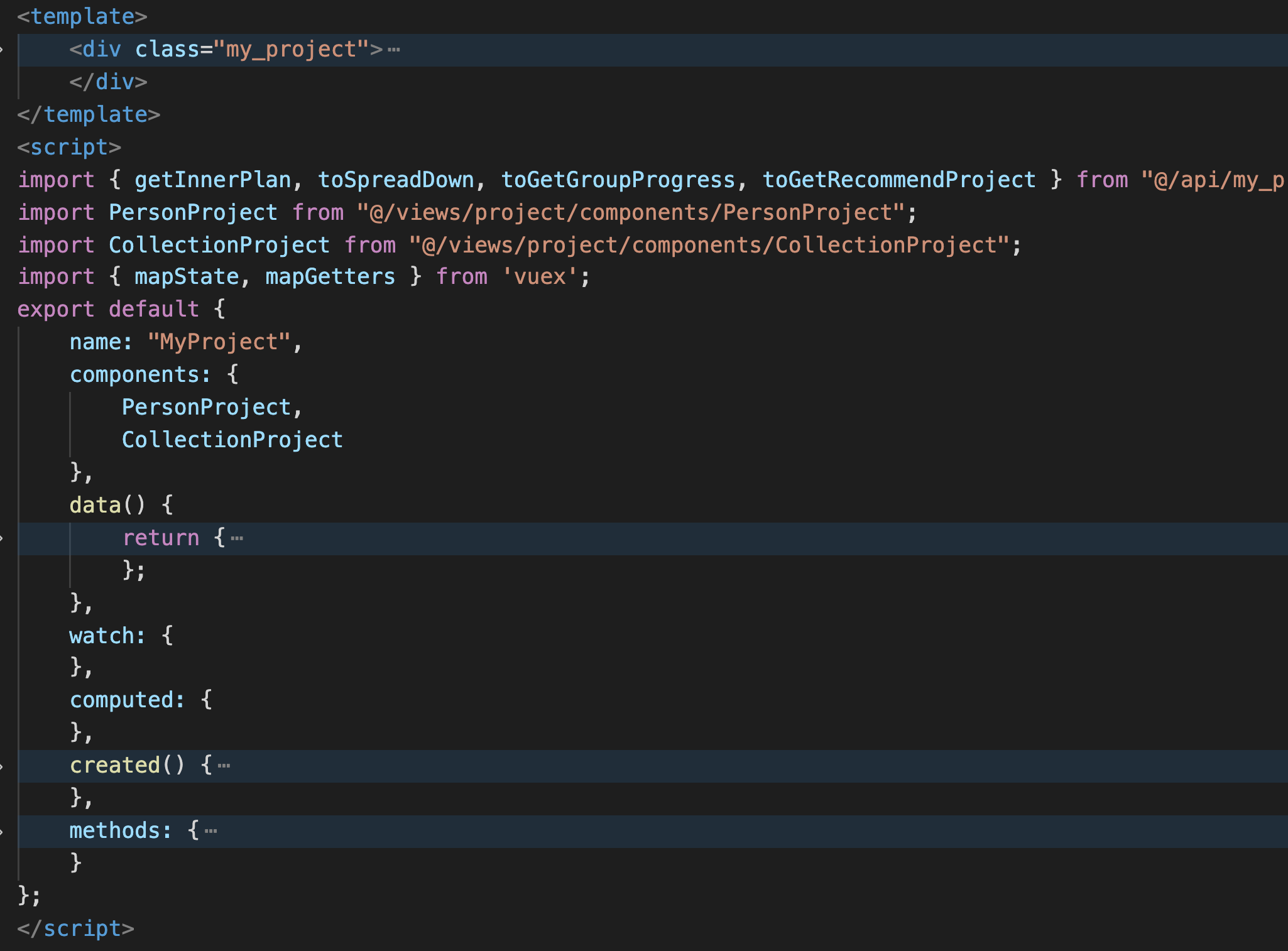
## 实验目的

可视化任务模块主要是为了让学生更加清晰的看到自己任务模块，以及各个任务的开发进度，以可视化的方式展示更加直观，并且更具可视化任务可以看到当前项木小组成员，，以及各个成员的开发进度，保证了项目以可预测的开发目标进行前进，并且结合在线IDE通过可视化任务列表，可以直接点击进入上次开发位置，提升整体工作效率

## 实验原理

## 1，组件封装

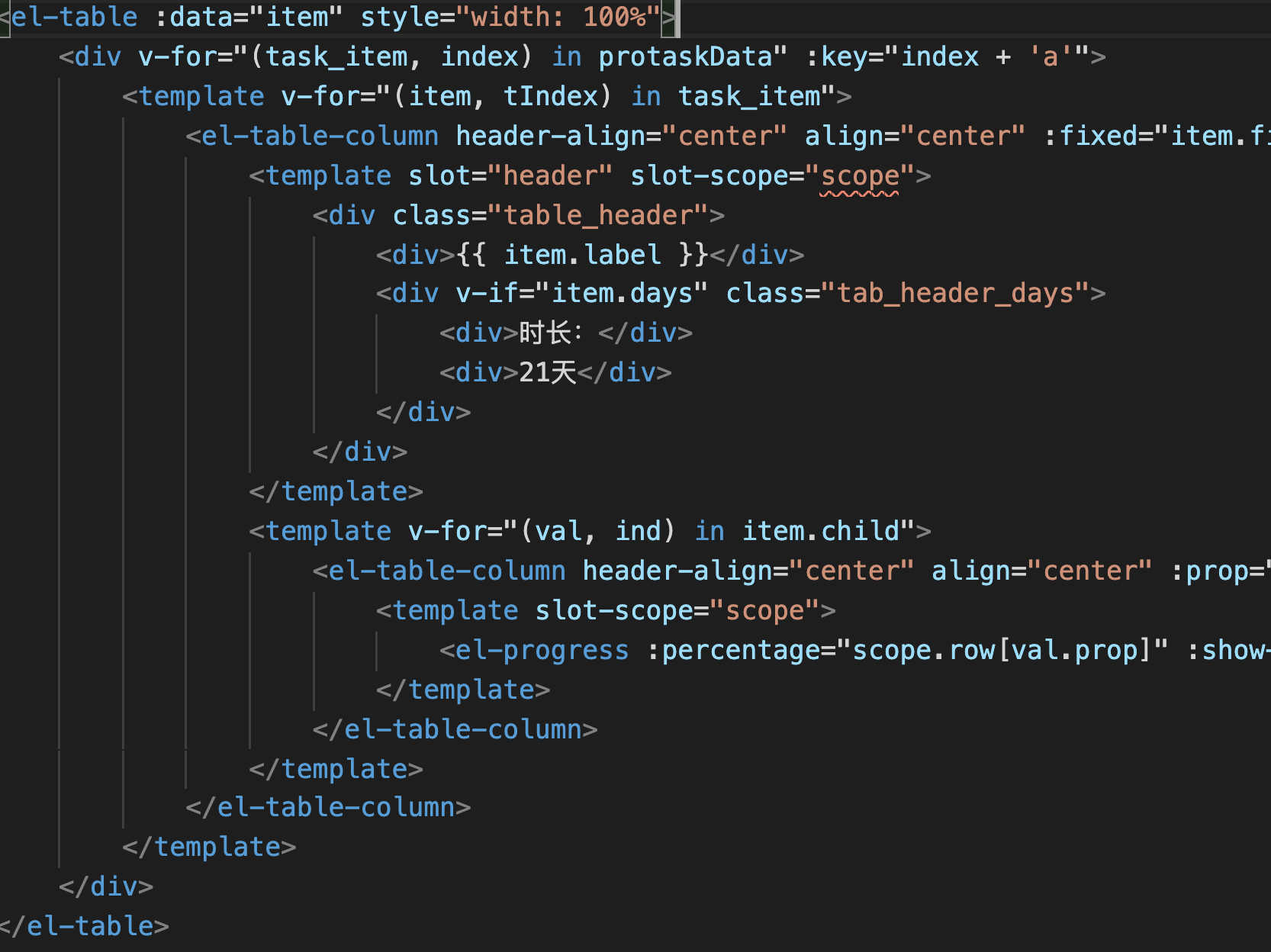
封装任务可视化列表组件，基于ElementUI进行二次的封装



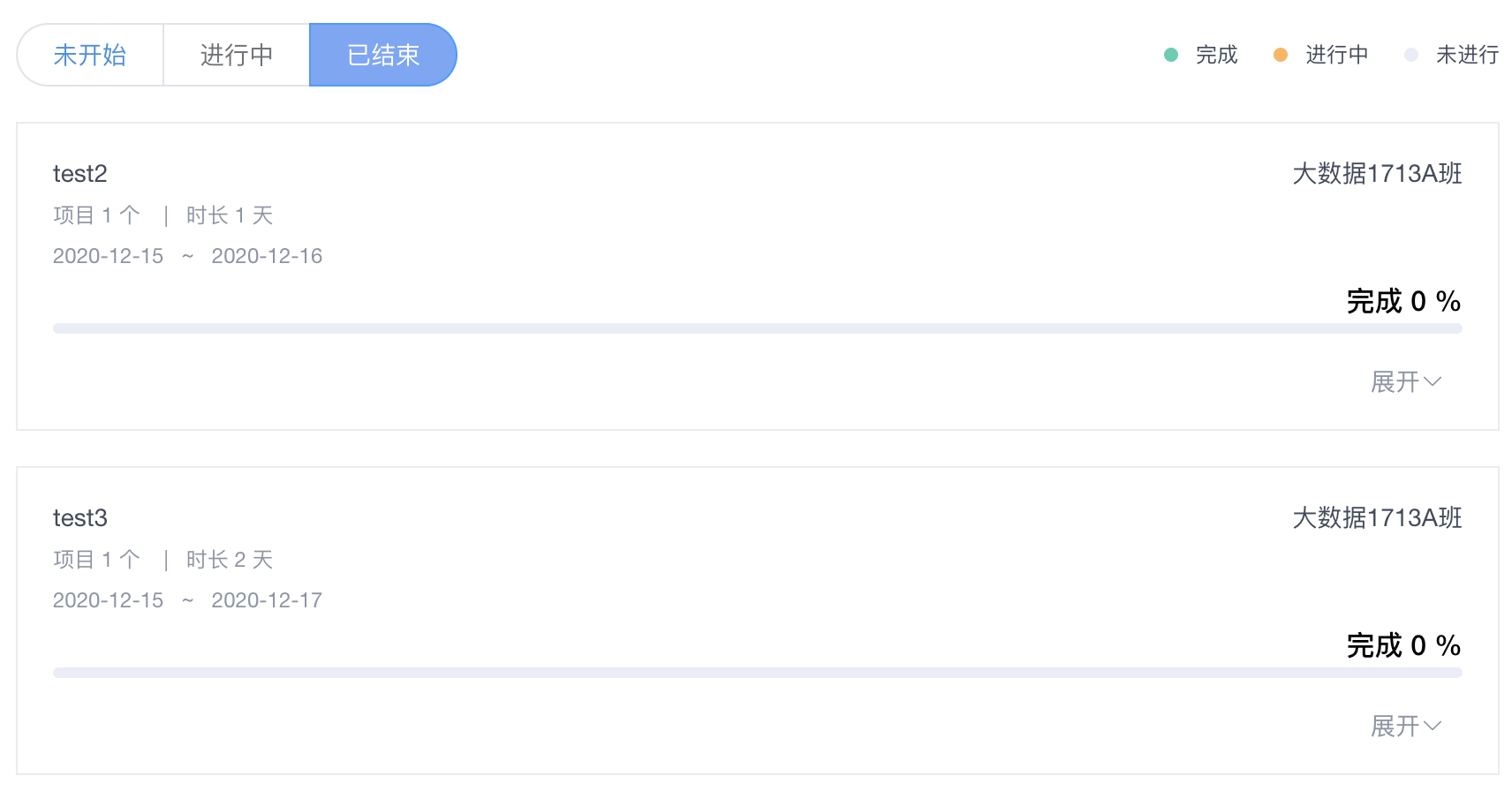
## 2，实现复杂表格配置

通过接口返回的动态任务列表，自动生成表格表头，并且输出列表，对应出每一个任务的进度





## 3，实现任务类型的切换

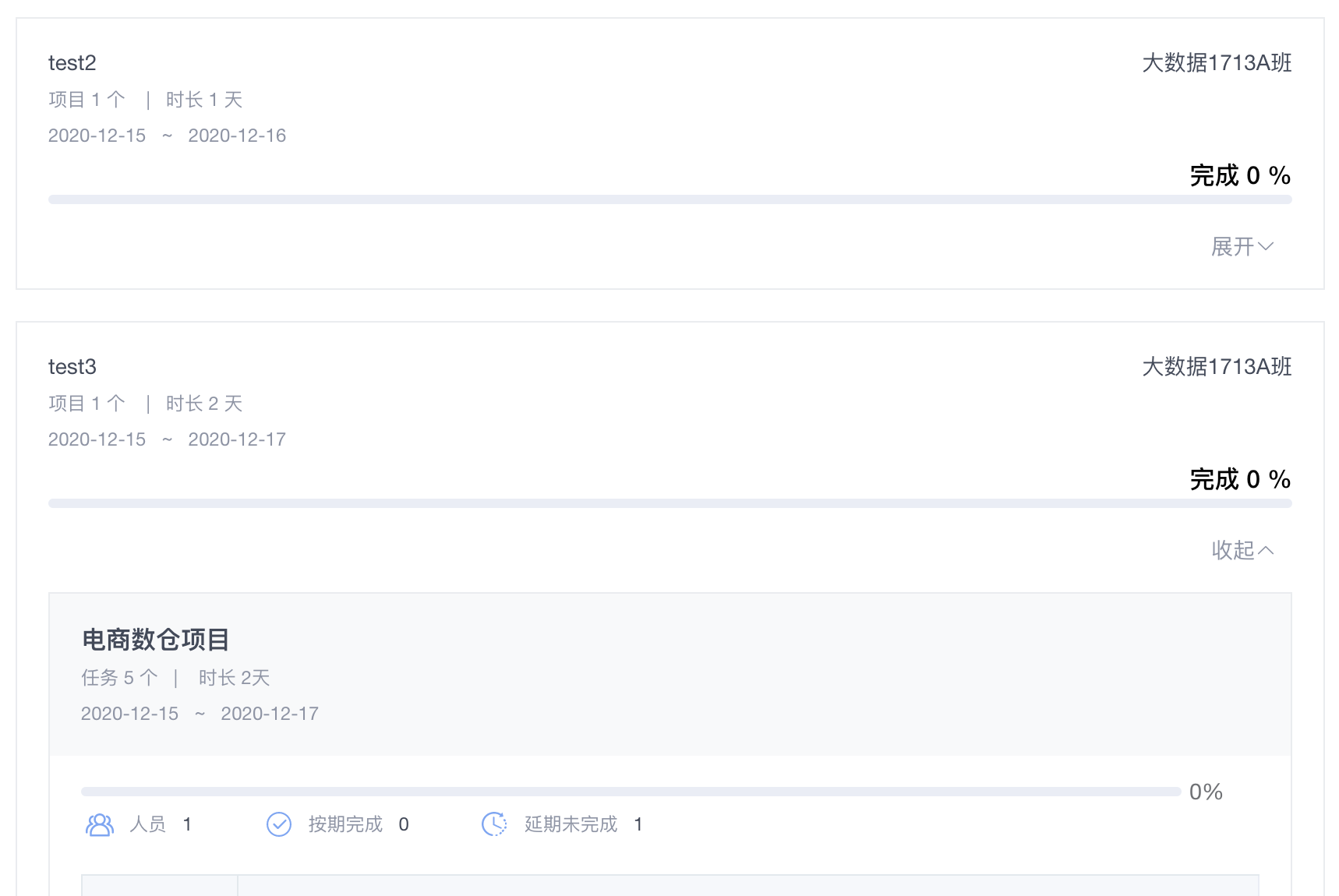


通过对任务类型的切换将（未开始，正在开发，已完成）的任务进行分类清晰展示不同的项目任务



## 4，实现任务详情的展示与收起

通过实现该功能进行任务详情的展示与收起，防止多项目任务的时候，可视化查看不清晰



## 实验环境

### 开发语言

Vue+Element-UI+Axios+Webpack+Nginx

### 开发环境

Node+npm

### 调试环境

Chrome Devtools中network审查，缓存测试

### Vue-cli配置

webpack配置调优，webpack构建脚本

### Nginx服务器

gzip压缩配置，服务器缓存配置

## 实验内容

### 页面布局

使用scss完成页面样式配置，修改elementUI样式匹配设计搞进行视图的百分白还原

### 组件调用

该功能模块需要调用

实现tab切换

el-tabs

el-tab-pane

实现进度展示

el-progress

实现表格

el-table

el-table-column

实现布局

el-row

el-col

### 接口调用

对接后端接口进行正确的数据渲染工作

getInnerPlan, 获取计划内项目

toSpreadDown, 获取任务

toGetGroupProgress, 获取小组开发进度

toGetRecommendProject 获取推荐项目

### 实训任务工作量对标公司工作量

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 组件封装 | 接口联调 | 页面布局 | 总计时 |
| 涉及的知识点 | 组件开发 | axios | css |  |
| 公司工作量 | 1天 | 1天 | 2天 | 4天 |
| 实训工作量 | .5天 | .5天 | 1天 | 2天 |

## 功能思路分析

## 关键技术点解决方案

无

### 实验步骤