

基于 React 的高校信息化项目管理系统的前端设计与实现

张新海¹ 蔡会霞¹
ZHANG Xinhai CAI Huixia

摘要

针对 MVVM 开发模式, 文章提出一种基于 React 环境的前端程序开发方法, 在分析比较主流前端开发框架后, 通过使用阿里的 Ant Design 框架, 结合 ahooks 组件库, 构建了完整的前端开发环境。以高校信息化项目管理系统的前端开发为例, 使用 TypeScript 作为开发语言, 在实现过程中使用 Ant Design 的各类组件实现系统布局和操作界面, 使用 useRequest 函数实现外部 API 接口数据管理, 使用 ahooks 的 useEffect 钩子函数实现数据状态监听和组件间的通信, 通过实践验证了高校信息化项目管理系统的前端程序实现方法。

关键词

项目管理系统; MVVM; React; Ant Design; ahooks

doi: 10.3969/j.issn.1672-9528.2025.07.012

0 引言

传统的系统开发通常采用前后分离 MVC^[1] 开发模式, 即后端实现业务逻辑, 前端通过模板引擎实现系统界面和用户交互。在前后分离的开发过程中, 往往是重后端轻前端, 系统的业务逻辑通常由后端决定, 随之带来了系统前端功能偏弱、用户体验较差、数据状态保持一致的问题。随着前端框架的迅猛发展, 开发模式进入到 MVVM^[2] 阶段, 即后端仅实现数据的增删改查, 前端在实现系统界面的同时还实现系统的业务逻辑。本文使用基于 React 环境的 Ant Design (Antd) 作为前端框架, 结合 ahooks 函数库, 设计与实现了高校信息化项目管理系统的前端程序。

1 前端开发框架可行性分析

1.1 主流前端框架分析

目前最主流的前端开发框架分别为 ElementUI 和 Antd^[3], 作为优秀的前端开发框架, 在组件库的数量上和功能上类似, 不同之处在于两者的开发环境和对后端数据的处理上。ElementUI 通过 Vue^[4] 实现业务逻辑, Vue 使用 JavaScript 作为编程语言, 通过 axios 组件库实现数据管理, ElementUI 可以适应市面上大部分系统的前端开发; Antd 通过基于 React 环境的 TypeScript 作为编程语言, 通过 useRequest 函数库实现数据管理, 并通过 ahooks 函数库实现数据状态监听和组件间的通信, 在数据更新上使用动态渲染^[5] DOM 元素的方法实现了更低的页面渲染。因此, Antd 在实现复杂业务逻辑的场景中更为合适。

1.2 实现系统界面的前端组件分析

在系统布局和界面设计上, 为提高开发效率, 应使用现有前端框架中的各类组件构建系统界面。通过分析高校信息化项目管理系统所需实现的功能, 通过使用 Antd 的 Layout 布局组件可以实现页面布局, 使用 Route 路由组件可以实现页面路由和菜单切换, 使用 Input、Button、Select、DatePicker、Modal 等组件可以实现数据编辑界面, 使用 Table、Pagination 等组件可以实现表格及分页界面, 使用 Steps 进度条组件可以实现项目进度管理界面。通过对系统的功能分析, 使用 Antd 的原生 UI 组件库可以实现系统中所有的界面。

1.3 实现系统业务逻辑的前端组件分析

通过对系统的业务逻辑进行分析, 主要的业务逻辑分为登录鉴权、数据列表、数据修改、数据渲染四大类。通过使用 Antd 的前置拦截器 RequestConfig 可以实现登录及鉴权功能, 通过使用 Antd 的 useRequest 函数管理外部 API 接口数据可以实现数据列表和数据修改的功能, 通过使用 ahooks 函数库^[6] 中的 useEffect 函数可以监听外部 API 接口数据的变化从而实现页面数据的动态渲染。通过对系统的业务逻辑分析, 使用 Antd 的各类函数库结合 ahooks 函数库可以实现系统中所有的复杂业务逻辑。

2 前端系统设计

2.1 前端系统功能设计

从信息化项目建设全过程出发, 将高校信息化项目^[7] 的建设过程分为立项、招标、合同、验收、付款五个阶段。其中每个建设阶段都需上传项目建设材料或文档, 如招标文件、

1. 江苏开放大学 江苏南京 210036

招标公告、合同文件、验收文件、付款发票等,项目建设进度根据每个阶段的项目文档完成比例决定。系统所需实现的总体功能分为3大类:一是根据项目名称查询项目的建设进度,并根据条件查询每个阶段的建设文档;二是根据部门查询部门所有的项目列表和文档,并自动得出部门所有项目的建设进度汇总;三是根据项目负责人查询该负责人所有项目列表和文档,并自动得出该负责人所有项目的建设进度汇总。

2.2 API 接口设计

系统采用前后分离设计模式,前端根据业务逻辑设计14个API接口^[8],前端程序通过调用后端API接口实现系统的业务逻辑。在系统API接口的设计和实现中,采用业务驱动的模式由前端设计API接口的数量、名称、参数以及调用方式。与传统的后端设计API接口的模式相比,前端设计的API接口更加精准和合理。API接口列表如图1所示。

接口地址	含义	调用方法
/api/login	用户登录	GET
/api/getProject, editProject	获取项目列表, 编辑项目	GET, POST
/api/getApproval, editApproval	获取立项, 编辑立项	GET, POST
/api/getBid, editBid	获取招标, 编辑招标	GET, POST
/api/getContract, editContract	获取合同, 编辑合同	GET, POST
/api/getAcceptance, editAcceptance	获取验收, 编辑验收	GET, POST
/api/getPayment, editPayment	获取付款, 编辑付款	GET, POST
/api/getAnalysis	获取项目进度	GET

图1 API接口列表图

2.3 前端系统架构设计

前端系统的架构设计为4个模块,分别为数据处理模块、全局状态数据模块、UI组件模块、数据监听模块。其中数据处理模块实现外部API接口数据的获取和更新;全局状态数据模块实现前端系统内部的数据管理;UI组件模块实现前端系统的布局和用户界面;数据监听模块实现前端系统的数据状态监听和组件间的通信。前端系统架构图如图2所示。

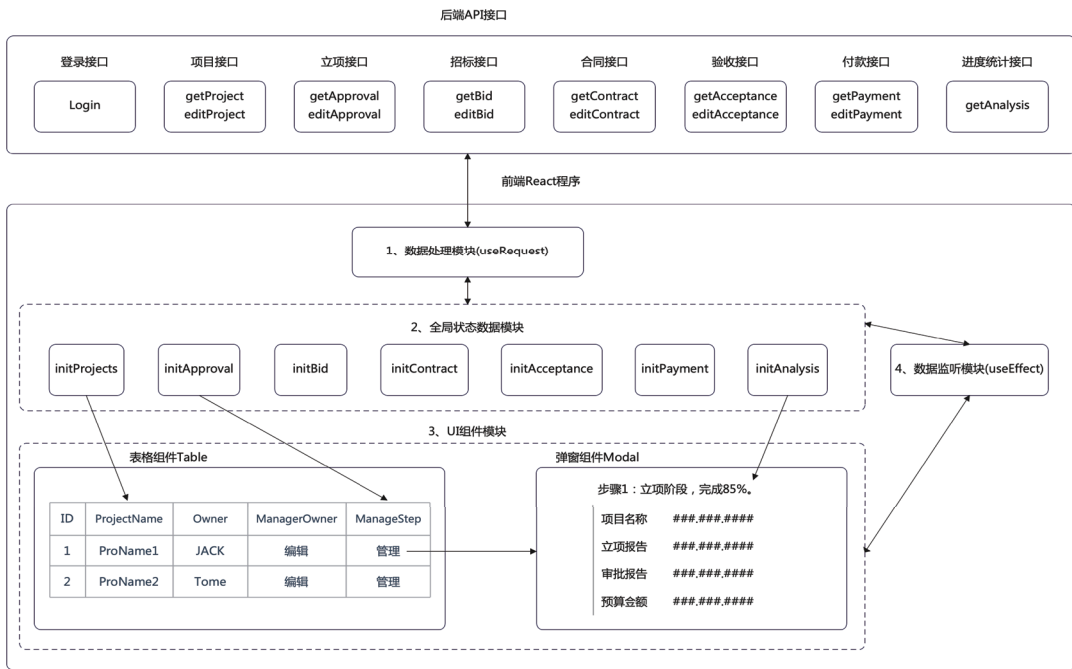


图2 前端系统架构图

(1) 数据处理模块

前端系统使用Antd的useRequest函数访问后端API接口实现数据的操作,useRequest函数通过GET请求方式获得数据,通过POST请求方式操作数据。通过在useRequest函数中添加参数队列Params实现复杂的数据查询与操作,Params参数队列中包含动态参数page、pageSize、sortQuery、serchQuery,参数队列通过useEffect钩子函数实现参数数量与参数值的管理。通过在useRequest函数的onSuccess方法中获取API接口返回的message消息参数,实现前后端在数据操作中的交互。

(2) 全局状态数据模块

前端系统将useRequest函数访问API接口获得的数据根据所需实现的功能依次注入图2中定义好的7个全局状态数据中,通过全局状态数据实现外部API接口数据到前端系统内部数据的转换,实现外部数据与前端系统内部数据的分层管理,当外部API接口的数据结构发生变化后只需配置前端系统的内部全局状态数据即可。其中全局状态数据分两类,一类是通过GET请求方式获得的列表数据或对象数据;一类是通过POST请求方式提交的被编辑的对象数据。

(3) UI 组件模块

该模块是构建系统布局和操作界面的模块,在实现过程中通过Antd的Route路由组件构建系统菜单,通过Table表格组件构建项目列表页面,通过Modal弹窗组件结合Form表单组件构建项目编辑页面,通过Setps步骤条组件构建项目建设步骤管理页面,通过Progress进度条组件构建项目建设中每个步骤的完成比例。

(4) 数据监听模块

该模块是实现前端系统业务逻辑的核心模块，数据监听模块通过 `useEffect`、`useUpdateEffect` 函数实现 3 个方面的功能：一是监听数据处理模块中 7 个全局状态数据的状态，当任何全局状态数据发生变化后，通过 `useUpdateEffect` 函数将变化后的数据动态触发渲染到 UI 组件中；二是监听前端系统中各类 UI 组件的操作状态，当组件操作状态发生变化后，通过 `useEffect` 函数调用相对应的 API 接口，将操作后的数据更新到相对应的全局状态数据；三是前端系统通过 `useEffect` 函数实现全局状态数据间的穿透传值和 UI 组件间的跨组件传值。

2.4 页面接口文件设计

页面接口文件的设计可以使前端程序在开发过程中使用面向对象的思维进行程序设计，通过将每个外部 API 接口数据在前端程序中转换成 TypeScript 格式的类型申明文件，在申明文件中详细描述接口数据的类型和属性，使每个 API 接口数据都具备相应的数据类型。如将项目列表数据在页面接口文件中申明为 List 列表类型，将项目数据属性中的项目名称、立项时间、所属部门、项目预算等属性分别申明为 String、Date、Number、any 类型，在前端系统的实现过程中可以通过页面接口文件的代码提示了解每个外部 API 接口数据的类型和属性，大大提高了代码效率。

3 系统实现

3.1 系统鉴权实现

根据前后分离开发模式，系统登录原理为 JWT 机制，用户通过 Login 接口登录后端发放给前端系统一个 Token 令牌，Token 中带有用户信息、系统权限和访问失效期。前端系统获得 Token 后通过 `localStorage.setItem` 方法将 Token 保存到本地，前端程序在后续访问 API 接口时将 Token 令牌带入 `httpHeader` 头，后端对 Token 进行鉴权后前端系统获得 API 接口的访问权限。在实现过程中，通过在 Antd 的 `request` 函数中编写前置拦截器 `RequestInterceptor`，确保在每次 HTTP 请求前带入 Token 从而实现前后端在分离状态下的鉴权。

3.2 项目管理的实现

项目管理功能主要使用 Antd 的 Table 表格组件和 Pagination 分页组件展示和管理项目列表，在表格组件中添加操作列，分别为添加、编辑、删除按钮。定义全局状态数据 `initProjects`，用于管理项目列表数据源。定义全局状态数据 `initAddProject`、`initDelProject`、`initEditProject`，分别用于项目的增删改操作。项目列表管理界面如图 3 所示。

ID	项目名称	项目创建人	创建时间	负责人	管理	操作
3	图书馆资源知识馆	admin	2023-07-18 09:42:00	admin	查看	添加 编辑 删除
4	开放教育数学教学平台升级项目	admin	2023-07-18 09:42:26	admin	查看	添加 编辑 删除
5	图书馆“智慧之星”电子图书数据库	admin	2023-07-18 09:43:30	admin	查看	添加 编辑 删除
6	江苏开放大学资源库平台功能测试迭代	admin	2023-07-18 09:42:44	admin	查看	添加 编辑 删除
7	教师业务系统二期	admin	2023-07-18 09:42:52	admin	查看	添加 编辑 删除
8	江苏开放大学智慧平台（教师端）维护	admin	2023-07-18 09:42:59	admin	查看	添加 编辑 删除
9	图书馆资源管理系统维护	admin	2023-07-18 09:43:04	admin	查看	添加 编辑 删除
10	江苏开放大学实验实训管理平台升级改造	admin	2023-07-18 09:43:17	admin	查看	添加 编辑 删除
11	江苏开放大学资源库平台数据迁移	admin	2023-07-18 09:43:24	admin	查看	添加 编辑 删除
12	江苏开放大学教师培训系统	admin	2023-07-18 09:43:30	admin	查看	添加 编辑 删除

图 3 项目列表管理界面图

(1) 项目列表的实现

使用 `useRequest` 函数访问 API 接口 `getProject`，将接口返回的数据注入全局状态数据 `initProjects` 中，将 `initProjects` 作为 Table 表格组件的数据源。参数中 `page` 和 `pageSize` 分别代表页码和每页记录数。`initProjects` 中的数据为 List 列表类型，List 中的每条数据为 Object 对象格式的数据，每个 Object 对象为一个完整的项目数据，如：`{ "title": "projectName", "dataIndex": "projectId", "key": "projectId" }`。通过将每个 Object 对象中的 `dataIndex` 属性与 Table 组件中的字段索引名逐一匹配后，生成完整的项目列表表格。在操作列中通过 Table 组件的 `Render` 函数，将行记录 `Record` 对象作为参数动态渲染出 3 个 Button 操作按钮，分别为添加、编辑、删除按钮，对应编写的执行方法分别为 `addHandle`、`editHandle`、`delHandle`。

(2) 项目编辑的实现

通过 Antd 的 Modal 弹窗组件结合 Form 表单组件实现项目编辑功能，在表格中点击编辑按钮时设置弹窗状态数据 `hideModal` 的值为 `true`，打开弹窗的同时将表格中当前行记录 `Record` 对象作为参数代入 `editHandle` 方法，在 `editHandle` 方法中运行全局状态数据 `initEditProject.run(Record.id)`，即将 `Record.id` 作为参数访问 API 接口 `editProject`，获得被编辑的项目对象数据后通过 `useEffect` 函数注入 Form 表单中进行编辑。编辑完成后，在 `initEditProject` 的 `onSuccess` 方法中调用 `hideModal` 方法关闭弹窗，并修改弹窗状态数据 `modalVisual` 的值为 `false`。

3.3 项目管理步骤的实现

以项目管理过程中立项管理步骤为例，通过 Steps 步骤条组件实现项目管理步骤的切换。定义 `onChange()` 方法，用于切换项目管理的五个步骤，`onChange=(value:number)=>{history.push()}`。运行全局状态数据 `initApproval.run(id)`，将项目的 id 号作为参数访问 API 接口 `getApproval` 获得被编辑的立项数据并注入 Form 表单组件中，编辑完立项数据后，调用 API 接口 `editApproval` 保存立项数据。完成立项数据编辑后，运行全局状态数据 `initAnalysis.run(id)`，获得立项步骤的进度数据，通过 Progress 进度条组件实现立项进度的实时完成比例。立项管理界面如图 4 所示。

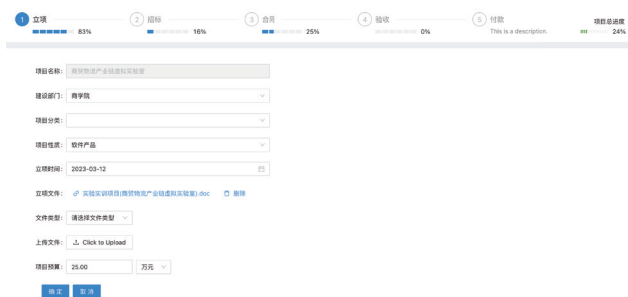


图4 立项管理界面图

3.4 数据状态监听的实现

(1) 表格状态监听

表格进行操作后,如何自动刷新表格,并且保持表格的页码和排序状态是最难实现的,用碎片化的JS技术^[9]实现会造成表格数据的紊乱。通过使用ahooks的useEffect函数,对表格的参数队列进行监听,当useEffect函数监听到参数队列中分页、排序、查询等参数发生变化后,自动运行全局状态数据initProjects.run(),重新获取表格数据源并按照表格操作前的状态重新渲染表格。

(2) 表格数据监听

对表格中某一行数据进行编辑后,一般通过获取表格全量数据,重新加载表格的方法实现表格数据的刷新,这种方法属于表格数据的重渲染,不仅效率低下而且会造成表格数据加载缓慢。通过使用useUpdateEffect函数,在表格首次加载时并不执行,而是在表格数据发生变化后,通过动态触发渲染的方法仅渲染发生改变的数据,极大提升了表格的渲染效率。

(3) 组件间传值

传统React通过props属性实现父子组件间数据的层层传递,不仅效率低下而且不能跨组件传值。以编辑表格中某个项目的功能为例,在表格中点击编辑按钮后,需要将当前行记录的值放入Modal弹窗的Form表单中,即需实现Table组件和Form组件间的跨组件传值。在表格中点击编辑按钮时运行全局状态数据initEditProject.run(Record.id),获得被编辑的行记录对象,通过useEffect函数对全局状态数据initEditProject进行监听,当发现initEditProject的值发生变化后,通过useEffect函数运行form.setFieldsValue(initEditProject.data),将被编辑的表格行记录值注入form表单中,实现了Table组件和Form组件间的跨组件传输,大大提高了组件间数据的传递效率。

3.5 前端系统的部署和上线

完成前端系统的全部功能开发后,通过运行node^[10]环境中的npm run build命令将前端程序编译打包成完整的前端代码,将前端代码部署在Nginx解析环境的80端口中,即可运行除API接口访问外的全部功能。通过配置Nginx反向代理

的proxy_pass的属性,在前端系统访问后端API接口时代理成访问前端系统自身Nginx环境的地址和端口,由前端系统的Nginx环境将每个API的访问通过proxy_pass代理成后端程序的域名和端口,即可实现前后端的跨域数据请求。

4 结语

本文从基于MVVM开发模式,通过前后分离开发模式,使用Antd的各类UI组件实现了高校信息化项目管理系统的系统布局和操作界面,使用Antd的useRequest函数管理外部API接口数据,使用React的全局状态数据管理前端系统内部数据,使用ahooks的useEffect钩子函数管理数据状态监听和组件间的通信,并通过以上技术实现了完整的信息化项目管理系统的前端代码,具体的代码思路 and 实现过程可供前端设计人员参考。

参考文献:

- [1] 欧阳宏基,李红,宋笑雪.一种基于ES2SH框架的Java EE应用架构[J].实验室研究与探索,2018,37(10):151-156.
- [2] 梁文婧,张宏海,张蕾蕾,等.基于MVVM模式的中国科技云门户管理系统的设计与实现[J].数据与计算发展前沿,2022,4(2):99-108.
- [3] 毛炎,任福,王功存,等.基于新型Web脚本样式框架构建城市规划编制信息平台:以ReactJS和Ant Design为例[J].测绘与空间地理信息,2017,40(8):81-84.
- [4] 康珊珊,刘莉,田凌,等.基于前端框架Vue.js的实验实训管理系统开发[J].实验室研究与探索,2023,42(3):281-284.
- [5] 孙卫真,孙星,向勇.前端渲染函数调用图工具的设计与实现[J].计算机工程与设计,2022,43(1):277-286.
- [6] 赵广强,凌捷.基于hook技术的进程管理系统研究[J].计算机工程与设计,2014,35(7):2325-2329.
- [7] 张新海,朱祎.基于MVVM模式SpringBoot框架的高校信息化项目管理系统的[J].信息技术与信息化,2024(5):54-58.
- [8] 周芯宇,陈伟,吴国全,等.REST API设计分析及实证研究[J].软件学报,2022,33(9):3271-3296.
- [9] 赵增敏,李惠敏.基于jQuery框架的Ajax应用开发[J].制造业自动化,2012,34(19):18-20.
- [10] 郭翠娟,暴宁,荣锋.基于MQTT的物联网平台研究与设计[J].计算机工程与设计,2022,43(8):2378-2384.

【作者简介】

张新海(1977—),男,江苏常州人,本科,高级工程师,研究方向:计算机软件开发,email:zhangxh@jsou.edu.cn.

蔡会霞(1973—),女,山西运城人,硕士,高级工程师,研究方向:计算机软件开发.

(收稿日期:2025-02-11 修回日期:2025-07-09)