

基于 Vue 框架的社交平台设计与实现

龙 源 黄媛媛

四川大学锦城学院计算机与软件学院 四川 成都 611731

【摘要】随着互联网行业日新月异的迅猛发展,以 Web 网页为基础的在线交互型娱乐已深入人们的日常生活,成为人们喜闻乐见的一种流行休闲方式。本文记录了:一种社交平台的设计与实现,该平台主要实现了内容编辑、内容分类提交、用户个人信息管理等功能。在开发中,利用 Vue 的 MVVM 特性实现平台数据的双向绑定,运用 HTML5 中的 localStorage 功能实现数据的本地存储从而达到理想的前端界面效果。

【关键词】网页开发; Vue; localStorage; MVVM

1 引言

信息时代的到来使人类生存的社会形态踏入以智能为主的信息社会,互联网的出现是人类社会历史上的一个新的转折点。所以,人民群众的生活、工作、娱乐等多方面的行为途径,就变得更加的便捷和丰富。信息传播是互联网时代的显著特点之一,各种类型的社交平台已成为人们记录、分享和获取信息的重要媒介 [1-2]。本文将介绍利用 Vue 框架及 localStorage 等浏览器相关技术栈设计与实现的社交平台。

2 技术栈

2.1 Vue 框架

Vue 是当下 Web 前端的三大主流框架 (Vue, Angular, React) 之一,用于构建用户界面。与其他大型框架相比,Vue 属于响应式轻量级框架,推行渐进式组件化开发,避免编程时出现重复的代码,使代码之间的耦合程度减小。

2.2 localStorage 浏览器 API

localStorage 是 HTML5 为浏览器新增的一个本地存储 API,可抽象为一个浏览器前端页面的数据仓库,主要作用是将数据长久地保存在客户端中。本 API 可供数据存放的容量一般为 5MB,数据仅在客户端中保存,而不会参与后端服务器的通信。

2.3 Node.js 平台和 npm 包管理工具

Node.js 是一个帮助 JavaScript 在服务端运行的平台,遵循非阻塞式 I/O 模型。npm 全称 Node Package Manager,是在部署 Node.js 环境时一起安装的模块管理分配工具,可以帮助开发者对需要的模块进行下载、安装,也可以用来管理已经安装的模块或者上传分享自己编写的包,帮助有需求的开发人员更加便捷地共享代码。

3 平台设计实现分析

3.1 功能需求分析

在互联网大数据时代,每天都会有各类信息以迅雷不及掩耳之势诞生着,所以帮助人们进行信息记录和对实时信息内容进行针对性分类获取是格外重要的,社交网站正是充当着这样一个促进信息传播的媒介角色。根据当下人们对社交网站功能的偏好来看,用户需要一个:集对外界时事动态进行获取,和发表个人见解、分享日

常及个人喜好等对生活内容进行记录的渠道。所以该平台主要实现对用户执行平台操作产生的信息进行记录,为用户提供多元化的社区信息内容。

(1) 主页登入/出:用户登录平台,可以通过手机号或者自定义的用户昵称和密码来验证私人信息,登录平台到达用户个人页面后也可以选择退出。

(2) 主页管理:用户可以在个人主页设置自己喜欢的用户名、头像并获取社区推荐话题、社区推荐发布动态内容等公开平台信息。用户可以通过对个性签名和自我介绍板块的编辑对自己的主页进行标记,可以通过留言板、说句话、推荐内容等版块记录用户自发性的信息共享。

(3) 分类推荐社区内容:用户可以获取由平台提供的分类社区模块内容的动态信息,如电影评分、音乐推荐和榜单排名等。主要版块内容分为音乐推荐、电影推荐等社区公共内容推荐模块。

3.2 实现可行性分析

需要对平台项目进行初始化构建,Vue CLI 脚手架是一个非常便捷的工具,构建过程中,于 Node.js 环境下在 cmd 控制台中执行 npm 命令语句,即可对项目执行:初始化、运行、添加插件安装包等操作。与此同时,Vue CLI 有对更改后的内容进行即时编译,报错时无法成功运行项目的特性,所以在编程过程中涉及到的错误与警告能够在控制台及时输出,提高开发效率。对于通过 JavaScript 实现逻辑流程中产生的错误,在运行过程中有时难以被捕捉,即可通过 vue-devtools 浏览器工具进行调试。

平台页面由多个 .vue 文件构成,每个文件对应构成一个或多个页面的独立或公共组件,实现组件的重用。在项目中添加 vue-router 和 vuex 安装包后,即可对页面组件内容进行切换,通过不同组件对公共状态执行修改更新操作。通过使用 localStorage,对用户执行平台操作产生的信息进行本地存储,使浏览器对页面进行持久记忆性渲染。

4 平台设计功能实现

4.1 功能模块设计

平台模块划分为安全管理、用户主页和娱乐推荐,具体交互功能的需求聚集在用户主页模块。其中,安全

管理模块中提供登录、注销等功能。文章管理中, 提供推荐、写日记等功能。用户主页管理中提供个性签名、介绍、留言板等功能, 如图 1。

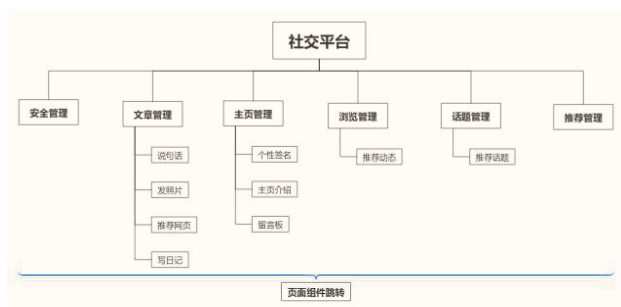


图 1 功能模块设计

4.2 平台框架搭建

在 Node.js 平台, 使用 Vue CLI 脚手架的 4.5.12 版本在 cmd 控制台通过 vue create 命令语句搭建项目。项目初始化搭建后, 在 cmd 控制台执行 npm i 模块名--save 命令语句, 安装项目所需要用的模块和包并将版本和状态记录在文件 package.json 中, 依赖包本身被存放在 node_modules 文件夹下。关键代码被编写在 src 文件夹, 图片等资源存放在 assets 中, 在 components 中定义各类组件。main.js 文件利用 ES6 的模块化引入模板, 是主页面配置的主入口^[3]。在 router 模块下配置项目路由。

4.3 功能模块实现

4.3.1 页面组件跳转

用户进入平台后, 需要通过登录操作到达用户主页, 从用户主页除了进行用户功能操作也可以选择退出平台或者到达其他娱乐推荐页面, 通过使用 Vue Router 路由管理器来实现组件的切换和页面的跳转。

在 router 目录下的 index.js 文件中通过 import 语句导入路由依赖包和需要配置的 .vue 组件文件。

配置 redirect 属性实现路由的重定向, 规定进入页面时首选的渲染组件。children 属性可进行多层嵌套。

注册时, 通过 children 属性定义当前组件的子组件实现组件的切换, 注册完毕后将路由 export 导出实现应用。

在被切换组件的父组件中, 使用 router-link 标签实现子组件的跳转, 使用 router-view 标签对切换组件进行页面渲染。

4.3.2 首页内容发布

在用户个人页面, 用户可以自由地记录和分享: 发表想说的话、分享喜欢的照片、推荐自己的喜好和日记描绘心情。

通过点击不同的链接可以切换不同操作指向的输入文本框。

在文本框中输入要发表的内容并点击发布按钮时, 文本框下方的发布列表会显示新增的发布内容, 包括自定义的昵称、头像、内容文字、发表时间和删除按钮。

在用户名后, 会标注用户选择的操作形式, 点击删除按钮, 当前选中的发布内容会从发布列表中删除。

当输入文本框中没有文字输入时则无法进行发布操

作, 此时点击按钮, 出现提醒用户输入内容的提示框。

在页面实现的过程中, 通过 localStorage 在浏览器建立一个缓存列表, 每发布新的内容时, 浏览器就将新增的内容存入缓存列表中, 使用 v-model 实现数据的双向绑定。当下一次打开页面或从其他组件切换回当前页面的时候, 浏览器便会判断 localStorage 中是否存在历史操作过程中录入的缓存, 有的话则读取历史缓存对页面进行记忆性渲染。执行语句部署在 mounted 钩子函数下, 页面挂载时执行。关键示例代码如下:

```
var releaseList = JSON.stringify(this.$store.state.releaseList);
localStorage.setItem("releaseList", releaseList);
localreleaseMseeage() {
  if (localStorage.getItem("releaseList") != "") {
    this.$store.state.releaseList = JSON.parse(localStorage.getItem("releaseList"));
  }
}
```

4.3.3 主页用户信息编辑

主页界面会展示用户的各类信息, 用户可以根据自己的个性签名和主页介绍进行编辑。在实现本页面的功能模块过程中, 也运用了浏览器 localStorage 缓存来实现内容的存取。

点击添加编辑按钮, 编辑处出现输入文本框, 在文本框中输入要添加的内容后点击修改或者保存按钮, 文本框消失且编辑处显示最新编辑内容, 即操作成功。

如果文本框中没有任何输入, 点击按钮后将保持为原本设置的默认文本。

在本界面设置留言板模块, 在文本框中写下留言, 点击留言按钮, 在下方的留言列表中显示留言内容、留言时间等信息, 执行删除按钮的操作可将留言列表中对应的留言记录删除。

4.3.4 社区内容推荐

平台在浏览发现部分和话题广场部分向用户提供社区动态和话题等内容的推荐, 以模板化列表的形式在页面排版呈现。在页面组件实例的 data 属性中编写列表数据, 固定模板下, 以数据驱动的形式通过 v-for 语句进行渲染, 同时给渲染元素加上一个 key 属性用来提高渲染性能, 使单模式大数据下出现的代码冗余现象得到了很好的改善。

4.4 实现界面示例



图 2 实现界面示例

5 结语

本平台采用了 Vue 框架和浏览器前端相关技术栈进行设计与开发, 最终呈现的平台界面已实现了上述的用户信息编辑记录、帮助用户获取多样化的社区信息内容等基本交互功能。经过开发测试证明, 本平台界面美观, 业务逻辑清晰, 交互功能丰富, 可以满足用户在平台进行社交和信息获取的基本需求, 向用户呈现了一种多元化的娱乐学习平台。目前本平台在安全性上考虑还不够, 比如对于用户名和密码在存储和传输过程中的加密强度问题, 在后续的工作中, 会对这方面需求继续完善, 力争提供一个界面简洁、安全性强的社交平台产品。

【参考文献】

- [1] 杨志乾. 网络信息分享行为的影响因素研究 [D]. 江汉大学, 2017.
- [2] 侯敏. 用户社交网络平台信息分享行为研究——以上海市大学生微博使用为例 [J]. 中国报业, 2020(10):16-17. 10.13854/j.cnki.cni.2020.10.007.
- [3] 胡雅丽. 基于 Vue.js 的“微商城”前端开发设计与实现 [J]. 电子技术与软件工程, 2020, (20):34-35.