

渐进式 JavaScript 框架 Vuejs 的全家桶应用

文/吕英华

摘要

本文的意义在于阐述 Vue 的全家桶能自动化构建、部署具有可维护性、性能稳定的项目，实现项目的模块化、组件化和规范化，且 Vue 的全家桶满足前端工程化的思想。Vue 是一款模块化、组件化的渐进式 JavaScript 框架，具有高性能和多用途的特点，可以创建可维护性和可测试性更强的代码库，可以在现有的服务端应用中加入 Vue 组件，带来更丰富的交互体验。

【关键词】Vue 模块化 组件化 全家桶 工具

实际项目中的最大挑战是前端业务日益复杂化和多元化促使项目周期快速演进迭代。随着时间的推移，业务需求的推进，项目的体系和形态发生了复杂的变化，会产生如下的问题：

(1) 最初的架构不再支持后续的变化，若在原有的架构进行迭代更新，会使代码更冗余，若重构整个项目，无法避免项目风险与资损。

(2) 因项目不断的迭代更新，开发周期加长，经多人开发和维护，出现代码质量低、可维护性差、可用性差等问题。

(3) 在多人协作开发时，没有统一的规范，无法进行高效的、高质量的协作开发。

以上问题可前端工程化的思想去解决。前端工程化本质上也是软件工程的一种，软件工程化主要关注可用性、可维护性、性能稳定性等方面。前端工程化包括模块化、组件化、规范化、自动化四个特点。模块化不等于组件化，模块化是对资源的拆分，组件化是在设计层面上或用户界面上的拆分。模块化是将一个大文件拆分成相互依赖的小文件，再进行统一的组装和加载；组件化是从用户界面上进行拆

分，拆分后的每个组件包含结构（HTML）、样式（CSS）和逻辑（JavaScript）；规范化是项目可持续发展的重要条件，其包括目录结构的制定、文档规范、编码规范、各种工具使用管理等内容；自动化是提高工作效率的重要条件，通过自动化工具完成持续集成、构建、部署、测试等工作。

本文工作的意义在于阐述 Vue 的全家桶能自动化构建、部署具有可维护性、性能稳定的项目，实现项目的模块化、组件化和规范化，且 Vue 的全家桶满足前端工程化的思想。

1 Vue全家桶的工具分析

在实际项目中，为了提高项目的即时效益，常常会降低项目的工程化去缩短开发周期，这会给后期的开发和维护增加难度。为了快速初始化一个具有前端工程化思想的项目，Vue 全家桶提供了快速开发项目的工具、状态管理工具、路由工具、请求工具及 UI 框架等。

1.1 快速开发项目工具

Vue CLI 是一个具有完全图形化界面和基于 Vue.js 进行快速开发的完整系统。Vue CLI 提供搭建、管理、开发交互式项目的手脚架；Vue CLI 服务是基于 Webpack 和 Webpack-Dev-Server 之上构建的，不仅能加载其官方提供的插件，还能加载其他 CLI 插件的核心服务；最重要的是，可以灵活调整每个工具的配置，无需 eject，节省配置时间。

1.2 状态管理工具

Vuex 是专门为 Vue.js 设计、以利用 Vue.js 的细粒度数据响应机制来进行高效的状态更新的状态管理库，其基本思想借鉴了 Flux、Redux 和 The Elm Architecture。Vuex 应用的核心是 Store（仓库），类似于一个容器，存

储该应用的大部分的状态（State）。Vuex 的状态（State）类似于全局变量，该变量提供给多个模块使用，但又区别于单纯的全局变量，因状态储存是响应式的，如修改了 Vuex 的某个状态，依赖于该状态的组件都会高效更新，且不能直接修改 Store 里的状态，需要显式提交给 Mutation。Vuex 的 Store 具有五个核心概念，分别为定义状态（State）、获取状态（Getter）、修改状态（Mutation）、触发 Mutation 的函数（Action）、将 Store 分割成模块（Module）。

1.3 路由工具

Vue Router 是 Vue.js 官方的路由管理器，是用来构建和管理 spa（单页面）应用的。传统的项目大多为多页面结构，即一个项目由多个 HTML 文件构成，在进行多页面跳转时，常受网络、性能等影响，会出现不同长度的空白页面，用户体验不好。单页面（spa）的核心思想为只有一个完整的页面，更新加载视图只需加载更新某个组件，不需要重新请求加载整个页面。Vue Router 通过改变路径实现组件之间的切换和跳转，并增加查询与设置路由参数、视图过度的效果、自定义滚动条行为等内容。

1.4 请求工具

一个完整的 Vue 项目需要向服务器请求数据，Vue.js 是基于 JavaScript 开发的，常用的 Ajax 也是可以使用的，在 Vue 2.0 版本之前，官方推出 Vue Resource，但之后不再维护更新，现官方推荐使用 Axios。Axios 是第三方插件，不仅能在 Vue 里使用，还能在 Nodejs、React 中使用。需要使用 Axios 请求的 Vue 组件，使用 import 导入即可。

1.5 UI框架

Vue 的 UI 框架分为 PC 端和移动端两

种, 可以根据项目需求选择合适的 UI 框架。

PC 端有饿了么出品的 Element、基于 Vue 开源的组件库 iview、三端样式一致的响应式 UI 库 muse-ui 等, 移动端有 Vux、饿了么出品的 Mint-UI、有赞出品的 Vant 等。每个支持 Vue 的 UI 框架都存在不极致的地方, 但丰富了 Vue 的 UI 环境, 开发者可以根据需求选择其一或多个 UI 库协同合作。

2 Vue全家桶应用分析

在使用 Vue 的全家桶搭建一个项目前需提前安装 Node、Npm 和 Webpack。

2.1 Vue CLI的应用

Vue CLI 只需 5 分钟即可创建并启动一个带热重载、保存时静态检查以及可用于生产环境的构建配置的项目。使用 Vue CLI 的前提是已安装 Webpack 和 8.9 版本以上的 Node。Vue.js 官方提供两个版本的 Vue CLI, 分别是 2.X 版本和 3.X 版本, 本文主要描述 3.X 版本的 Vue CLI 应用。

(1) 全局安装手脚架, 初始化项目所需的环境: `npm install -g @vue/cli`。安装完之后通过 `vue --version` 检验是否安装成功。手脚架一般具有自动构建代码、本地开发与调试, 热替换与热加载、CSS 模块化等功能。

(2) 构建一个新项目, 可以选择默认的 preset 和手动选择的 preset: `vue create my-project`。手动选择配置面向生产的项目, 提供更多的选项, 目前手动选择设置提供了添加依赖、单独配置文件、保存当前配置等配置内容。Vue 还提供了 `vue ui` 命令以图形化界面创建和管理项目。

(3) 基于项目需求安装依赖: `npm add **`。以此命令安装的插件会在 `package.json` 文件中列出来, 以查看安装情况和项目移动后一键安装所有插件使用。

(4) 启动开发服务器并附带开箱即用的模块热重载: `npm run serve`。启动服务器后默认打开浏览器窗口, 端口号为 8080。Vue 提供两种方式修改服务器的配置, 一是通过命令行 `vue-cli-service serve [options] [entry]` 修改, 二

是通过修改 `vue.config.js` 文件的配置。

(5) 打包编译程序: `npm run build`。此命令会在 `dist/` 目录下产生一个可用于生产环境的包, 带有 JS/CSS/HTML 的压缩和为缓存而做的自动的 vendor chunk splitting, 包的 chunk manifest 内联在 HTML 文件里。

2.2 Vuex的应用

一个应用的前端页面可以千变万化, 但数据一直是支撑应用的最重要部分。在 Vue 应用中, Vuex 就充当了数据提供者的角色, Vue 则只需要关注页面的展示与交互。Vuex 的 Store 具有五个核心概念, 分别为 State、Getter、Mutation、Action、Module, 分别分析每个功能。

(1) State: 存储状态 (变量)。只能访问状态, 不能修改状态。当 Store 实例注入到根组件下, 如需访问某个状态时子组件通过 `this.$store` 访问, 如需访问多个状态, 可以通过 `mapState` 辅助函数生成计算属性。

(2) Getters: store 的计算属性, 用来修饰状态, 并不修改状态。

Getters 的第一个参数为 state, 第二个参数可以为其他 Getter。子组件可以通过 `store.getters` 对象访问。

(3) Mutation: 修改 Store 状态的唯一途径。Mutation 类似 JavaScript 中事件, 具有事件类型和回调函数, 子组件可以通过 `store.commit('type', parameter)` 调用。

(4) Action: 类似 mutation, Mutation 可以直接修改 Store 状态, Action 需要调用 Mutation 的方法去修改 Store 的状态, 属于异步操作。

(5) Module: 当状态存储过多时, Store 会变得臃肿, 故 Module 可以将 Store 对象分割成多个模块, 每个模块拥有自己的 State、Mutation、Action、Getters。

2.3 Vue Router

前端路由的核心在于改变视图的同时不向后端发出请求, 浏览器提供了两种方式去实现, 分别为 hash 和 history。Hash, 即 URL 中

的 # 符号, 如 `http://www.abc.com/#/abc`, hash 值为 `#/abc`。虽然 URL 带有 hash 值, 但是不会被带入 HTTP 请求中, 因此改变 hash 值不会重新加载页面。History 利用了 HTML5 History Interface 中的 `pushState()` 和 `replaceState()` 方法, 应用于历史记录栈, 具有对历史记录修改的功能。Vue Router 利用了这两种方式实现前端路由跳转, 其默认为 Hash 模式, 如果使用 History 模式, 需要后端配置支持。

3 总结

Vue CLI 是 Vue 的手脚架工具, 可以自动生成项目目录, 配置 WebPack, 安装各种依赖包, 管理各种资源, 具有前端工程化的自动化特点; 通过 Vue 构建单页面的应用, 用 Vue Router 对页面进行局部无刷新的替换, 用 Vuex 对应用中的各种状态进行管理, 具有前端工程化的模块化、组件化的特点; Vue 的全家桶具有完善的功能和 API 文档, 给 Vue 的开发者提供规范的开发环境, 这就是前端工程化的规范化特点。这就是 Vue 的全家桶可以解决因前端业务日益复杂等原因引起的问题。

参考文献

- [1] 柴青山. 基于 MVVM 模式的 Vue.js 框架在物流软件自动化测试系统中的应用研究 [D]. 北京邮电大学, 2019.
- [2] 汪彤. 基于 Node.js 的图书共享平台的设计与实现 [D]. 北京邮电大学, 2018.
- [3] 麦冬, 陈涛, 梁宗涛. 轻量级响应式框架 Vue.js 应用分析 [J]. 信息与电脑 (理论版), 2017 (07): 58-59.

作者简介

吕英华 (1993-), 女, 广西壮族自治区钦州市灵山县人。大学本科学历。助教。研究方向为软件开发。

作者单位

北海职业学院电子信息工程系 广西壮族自治区北海市 536000