后台管理系统选型背景:

为了实现前后端分离,前端只负责调用后端接口,实现数据的展示,所以不在考虑使用jsp的形式来作为前台展示的技术手段。Jsp的耦合度比较大，所有的数据展示都是通过对象的形式绑定到页面来展示的。

vue是一个兴起的前端js库，是一个精简的MVVM。从技术角度讲，Vue.js 专注于 MVVM 模型的 ViewModel 层。它通过双向数据绑定把 View 层和 Model 层连接了起来，通过对数据的操作就可以完成对页面视图的渲染。当然还有很多其他的mvmm框架如Angular，React都是大同小异，本质上都是基于MVVM的理念。 然而vue以他独特的优势简单，快速，组合，紧凑，强大而迅速崛起

 vue和jquey对比

jQuery是使用选择器（$）选取DOM对象，对其进行赋值、取值、事件绑定等操作，其实和原生的HTML的区别只在于可以更方便的选取和操作DOM对象，而数据和界面是在一起的。比如需要获取label标签的内容：$("lable").val();,它还是依赖DOM元素的值。

Vue则是通过Vue对象将数据和View完全分离开来了。对数据进行操作不再需要引用相应的DOM对象，可以说数据和View是分离的，他们通过Vue对象这个vm实现相互的绑定。这就是传说中的MVVM。

综上所述，使用Vue作为后台管理系统的前端架构。

相关架构选型：

Vue-cli

webpack

vue2 + vuex + vue-router + webpack + ES6/7

●vue2：用户界面框架，前端主要业务通过它的语法来编写。

●vuex