

三峡大学

计算机与信息学院

《 软件测试课程设计 》

报告

开课学期： 2023 年 春季 学期

网选班号： 2 班

学 号： 2020162117

姓 名： 钟婉莹

完成日期： 2023 年 6 月 4 日

使用 Cuketest 框架设计自动化 UI 测试

钟婉莹

摘 要： 本文使用 Cuketest 框架对自研项目：二手汽车后台管理系统进行自动化 UI 测试，即 Web 自动化测试，借助 Web 录制工具录制了登录注册 admin 账户，并输入汽车类型，动态，用户信息等数据，更改后查看主页 echarts 图的变化，并与在线用户 zhang 进行聊天等 recording.js 脚本文件，并学习用 Cuketest 框架进行了 ERP 订单系统的参数化和数据驱动，添加检查点和截屏等功能。进一步提升了我对 Cuketest 框架的理解和掌握。借助 Cuketest 框架最后生成的报告实现了测试数据的可视化，并通过生成报告的截屏以及剧本 feature 文件可以很清晰地进行项目的 BDD 测试，使得测试结果更加直观清晰。但同时作为一个比较新的框架，我也在使用 Cuketest 框架中感受到了一些问题。

关键词： Cuketest 框架；BDD 测试；自动化 Web 测试；Web 录制；剧本文件；数据可视化

目录

1. 概述	3
1.1.Cuketest 框架简介	3
1.2. Web 自动化测试简介	4
2.任务概述.....	5
2.1 目标.....	5
2.2 测试环境.....	5
2.3 条件与限制.....	5
3.测试计划.....	6
3.1 测试范围.....	6
3.2 测试目标.....	6
3.3 测试类型.....	7
4.测试内容及方法.....	8
4.1 测试数据准备	8
4.2 测试内容.....	9
4.2.1 脚本文件的书写	9
4.2.2 场景对应项目运行 UI 截图	15
4.2.3 测试报告及通过率.....	26
5.结果分析.....	28
5.1 整体 UI 自动化测试结果分析	28
5.2 对 Cuketest 框架使用感受的总结:	28
5.2.1 CukeTest 的优点.....	28
5.2.2 CukeTest 的缺点.....	29
参考文献.....	33

1. 概述

本次软件测试结课作业我选择的是，使用基于 Cuketest 框架对自研项目：二手汽车后台管理系统以及官方提供案例 ERP 订单自动录入系统设计自动化 UI 测试

1.1.Cuketest框架简介

CukeTest 是开发测试自动化脚本的一个灵巧方便的工具。用户可以使用此工具快速创建 BDD（行为驱动开发, Behavior Driven Development）测试脚本。它集成了 Cucumber 框架和 JavaScript，可视化编辑，调试功能，并有多数 html 报告模板可供选择。同时它能够实现了各种类型应用的自动化，包括 Web、Windows、移动设备、API 等等。Apache jmeter 可以用于对静态的和动态的资源（文件，Servlet，Perl 脚本，java 对象，数据库和查询，FTP 服务器等等）的性能进行测试。它可以用于对服务器、网络或对象模拟繁重的负载来测试它们的强度或分析不同压力类型下的整体性能。你可以使用它做性能的图形分析或在大并发负载测试你的服务器/脚本/对象。

这里是一些主要特点：

提供可视化的方式来编写 BDD 测试用例和脚本，使用 CukeTest 用户可以专注于内容创建，而不是 BDD 文档的格式，用户只需了解 BDD 的一些基本概念即可使用它。CukeTest 工具内置了丰富的可视化操作，可以在整个脚本创建过程中为用户提供贴心的提示和指导。

通过单击步骤(Step)生成步骤定义(Step Definition)代码，从步骤文本轻松跳转到步骤定义代码，反之亦然。

提供多种方式来执行您的测试，并可以配置标签过滤和不同种类的浏览器来自定义执行。另外，场景可以在编辑时进行验证和运行。

帮助你管理代码和 BDD 测试用例文档，在文档和代码之间进行精准的匹配和跳转。

剧本（gherkin）文件编辑可以在可视模式(Visual Mode)或文本模式(Text Mode)。多种语言支持功能文档。您可以在 CukeTest 中将文档从一种语言格式转换为另一种语言格式。

可视化模式下轻松编辑测试数据，还可以将数据导入到 gherkin 文件或从中导出数据。

CukeTest 可在测试运行之前全面验证您的脚本，将执行过程中可能出现的问题降至最低。

全面的 Windows 应用自动化功能，可以针对各类 Windows 应用实现自动化，包括 Qt、Windows 原生应用、.NET、Delphi、DevExpress 等类型的应用都提供支持。提供方便易用的应用对象模型库管理功能、强大的控件识别功能，拖拽生成代码，方法调用测试等功能，能够快速生成 Windows 自动化脚本。

1.2. Web自动化测试简介

CukeTest 提供了完善的 Web 自动化测试,除了手动编写 Web 自动化脚本的环境(Node.JS 环境、测试样例、包管理器),另外还提供了自动生成 Web 自动化脚本的功能——即录制和回放 Web 操作的功能。

Web 自动化测试主要有以下优点:

提高测试效率: 自动化测试可以替代手动测试,节省时间和人力成本,可以对大规模、繁琐的测试任务进行快速、准确、可靠的测试。

提高测试准确性: 自动化测试可以通过比较测试结果与预期结果来检查是否存在错误或异常。

节省开发成本: 自动化测试可以在开发过程中及早发现和修复问题,节省成本并提高开发效率。

支持跨浏览器和跨平台测试: 自动化测试工具可以模拟多种浏览器和操作系统环境,进行全面、细致的测试。

总的来说,Web 自动化测试为测试人员和开发人员提供了一个更高效、更准确、更全面的测试方式,同时也提高了软件的质量和可维护性。

2.任务概述

2.1 目标

通过 CukeTest 测试框架，将二手汽车后台数据自动录入并检验相关功能如：登录注册 admin 账户，并输入汽车类型，动态，用户信息等数据，更改后查看主页 echarts 图的变化，并与在线用户 zhang 进行聊天，同时根据剧本的编写以及生成报告记录发现的缺陷、修改缺陷，分析缺陷产生原因。

2.2 测试环境

硬件环境：Windows 11 AMD Ryzen 7 6800H with Radeon Graphics 3.20 GHz 16.0 GB

软件环境：CukeTest 1.7.11.1

2.3 条件与限制

硬件环境：服务端 CPU 2GHz 以上 内存 2G 以上；硬盘剩余空间不低于 50G； 客户端手机 1G 内存及以上；硬盘空间 40M 及以上

软件环境：服务端 操作系统 CentOS 7.3 64 位 手机操作系统 Android5.0 及以上或 ios9 及以上 PC 端浏览器 Google Chrome 79.0.3945.117（64 位）

3.测试计划

3.1 测试范围

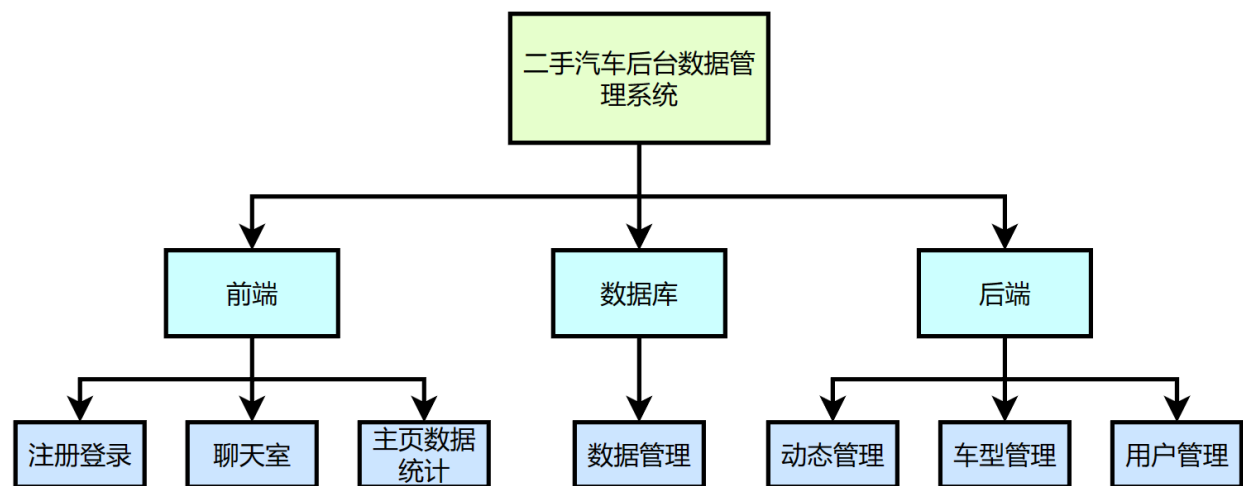


图 1 二手汽车后台数据管理系统功能图

本次我们主要进行的是 BDD 测试，由于项目基于 Spring boot+Vue+MyBatis-Plus，因此此次 Web 自动化测试同时涉及前端后端和数据库。同时在 Web 自动化的过程中属于黑盒测试，但由于该项目为自研项目，因此后续根据测试情况修改了部分代码。

3.2 测试目标

表格 1 测试目标表

类别	目标
进度目标	系统测试完成日期：2020/06/4
执行目标	测试用例覆盖率 100%
	测试用例通过率 100%
	缺陷清除率 100%

3.3 测试类型

表格 2 测试类型表

序号	测试类型	测试目的	是否采用	备注
1	功能性测试	根据系统需求文档和设计文档，检查产品是否正确实现了功能	√	
2	UI 测试	检查界面是否美观	√	
3	兼容性测试	在不同浏览器上能正常运行	√	
4	流程测试	按操作流程进行的测试，主要有业务流程、数据流程、逻辑流程、正反流程，检查软件在按流程操作时 是否能够正确处理	√	
5	回归测试	检查程序修改后有没有引起新的错误、是否能够正常工作以及能否满足系统的需求	×	
6	性能测试	提取系统性能数据，检查系统是否满足需求中所规定达到的性能	×	
7	接口测试	检查系统能否与外部接口正常工作	×	
8	安全性和访问控制测试	应用程序级别的安全性：检查用户只能访问其所属用户类型已被授权访问的那些功能或数据。 系统级别的安全性检查只有具备系统和应用程序访问权限的用户才能访问系统和应用程序。	×	

4.测试内容及方法

4.1 测试数据准备

此次自研项目为二手汽车后台数据管理项目，因此数据格式与内容是基于数据库进行增加和修改的，如下图：

id	username	password	nick_name	age	sex	address	role
1	admin	admin	管理员	21	女	湖北省	1
13	zhang	123	张三	20	男	湖南省	2
14	wang	123	王梦晨	24	女	江苏省	2
15	li	123	李雪	22	女	云南省	2
16	qian	123	钱江	22	男	四川省	2
18	zhong	123456	小钟	18	女	江西省	1
22	zhou	123456	周	18	男	陕西省	1
40	xiaohong	123456	nick	30	女	三峡大学	(Null)
45	11	123456	11	11	男	三峡大学	(Null)

图 2 user 表

id	name	price	author	create_time	cover	user_id
34	Jeep	227000.00	<p>此车购买三年</p>	2022-08-24 10:36:33	(Null)	19
35	江淮	139000.00	<p>此车购买三年</p>	2022-08-24 10:36:58	(Null)	20
36	捷豹	151000.00	<p>此车购买五年</p>	2022-08-24 10:37:18	(Null)	22
37	长安	179000.00	<p>此车购买四年</p>	2022-08-24 14:05:48	(Null)	1
39	奔驰	500000.00		2023-06-03 00:00:00	(Null)	1
164	宝马	500000.00		2023-06-04 20:57:12	(Null)	1

图 3 usedcar 表

id	title	content	author	time
4	各位用户谨防诈骗	<p>444444</p>	管理员	2021-07-17 18:55:27
5	有什么问题请及时咨询	<p>7777</p>	管理员	2021-07-17 19:14:14
6	今天喜提一辆二手宝马,	<p>dasdawdad</p>	李雪	2022-08-22 17:52:39
7	二手车怎么还卖这么贵	<p>wqeadwa</p>	李雪	2022-08-22 18:42:32
8	和二手车卖家商量了一	<p>wqdwad</p>	周	2022-08-24 11:06:57
13	这车开着不错	<p>9成新买的, 好好	张三	2023-06-04 09:25:40
14	请各位用户有什么问题	<p>123</p>	管理员	2023-06-04 09:43:47
33	今天喜提宝马, 开的真		管理员	2023-06-04 20:47:46

图 4 动态表

4.2 测试内容

4.2.1 脚本文件的书写

CukeTest 是一个强大的脚本文件(或特性文件)编辑器。脚本文件后缀名为*.feature。在编辑脚本文件时，它提供了 2 种编辑模式，可视模式和文本模式。而我在进行 Web 录制自研项目的功能以后，通过脚本也就是 feature 文件对于我想测试的功能进行了编写，具体测试方案如下：

此为文本模式：

language: zh-CN

功能：<二手汽车后台数据自动录入并检验相关功能>

登录注册 admin 账户，并输入汽车类型，动态，用户信息等数据，更改后查看主页 echarts 图的变化，并与在线用户 zhang 进行聊天

场景：注册 admin 账户

当浏览到注册网页 "http://localhost:9876/register"

那么输入用户名"admin"，密码"admin"，验证码"0000"，回到登录页面

场景：进入二手汽车后台数据管理系统

当浏览到登录页面 "http://localhost:9876/"

那么输入用户名"admin"，密码"admin"，确认密码"admin"，登录账号导航到主页

场景：新增汽车类型相关信息

当浏览汽车类型 "http://localhost:9876/"

那么点击"新增"按钮，输入品牌"特斯拉"，价格"400000"，点击"确定"按钮，点击"新增"按钮，输入品牌"特斯拉"，价格"500000"，点击"确定"按钮

场景：模糊查询汽车类型相关信息

当输入"特斯"模糊查询特斯拉车型

那么验证是否查询成功

场景：批量删除汽车类型相关信息

当删除"特斯拉"车型

那么验证是否删除成功

场景：编辑修改汽车类型相关信息

当点击"编辑"按钮修改"奔驰"价格"500000"，再修改回"540000"

那么验证是否修改成功

场景：新增动态相关

当跳转到动态网站 "http://localhost:9876/news"

那么点击"新增"按钮，输入标题"今天喜提宝马，开着很爽"，点击"确定"按钮

场景：新增用户相关信息

当浏览到网地址 "http://localhost:9876/user"

那么点击"新增"按钮，输入用户名"11"，昵称"11"，年龄"11"，性别选择"男"，地址"三峡大学"，点击"确定"按钮，点击"新增"按钮，输入用户名"22"，昵称"22"，年龄"22"，性别选择"女"，地址"三峡大学"，点击"确定"按钮

场景：到主页查看 echarts 图在修改用户数量后的地域比例改变

当跳到搜索网站 "http://localhost:9876/home"

那么查看 echarts 图是否发生改变

场景：登录普通用户"zhang"

当跳到登录网页 "http://localhost:9876/"

那么输入用户名"zhang"，密码"123"，验证码"0000"，导航到主页

场景：与"admin"发起在线聊天并在聊天结束后关闭页面

那么跳转到 "http://localhost:9876/im"，点击"气泡"与"admin"发起聊天，admin 发送信息"你好有什么需要帮助的吗"，输入"请问可以跟我说一下平台支付步骤吗？"，并点击"发送"按钮，关闭页面

下面通过可视模式对具体的测试行为进行描述：

场景 1：进入注册页面注册 admin 账户，并跳转到登录页面

1. 场景：注册admin账户

1. 当 浏览到注册网页 "http://localhost:9876/register"

2. 那么 输入用户名"admin"，密码"admin"，确认密码"admin"，回到登录页面

图 5 场景 1

场景 2：进入登录页面输入用户名，密码，验证码，登录后跳转到项目主页

2. 场景：进入二手汽车后台数据管理系统

1. 当 浏览到登录页面 "http://localhost:9876/"

2. 那么 输入用户名"admin"，密码"admin"，验证码"0000"，登录账号导航到主页

图 6 场景 2

场景 3：进入汽车类型页面以后新增汽车类型信息，如下表：

表格 3 添加汽车类型信息表

品牌	价格
特斯拉	400000
特斯拉	500000

3. 场景：新增汽车类型相关信息

1. 当 浏览汽车类型 "http://localhost:9876/"

2. 那么 点击"新增"按钮，输入品牌"特斯拉"，价格"400000"，点击"确定"按钮，点击"新增"按钮，输入品牌"特斯拉"，价格"500000"，点击"确定"按钮

图 7 场景 3

场景 4：模糊查询“特斯”，验证是否查询成功

4. 场景：模糊查询汽车类型相关信息	▶ ▼
1. 当 输入"特斯"模糊查询特斯拉车型	>
2. 那么 验证是否查询成功	>

图 8 场景 4

场景 5：批量删除特斯拉车牌信息

5. 场景：批量删除汽车类型相关信息	▶ ▼
1. 当 删除"特斯拉"车型	>
2. 那么 验证是否删除成功	>

图 9 场景 5

场景 6：编辑修改汽车车型相关信息，并查看详情页面

6. 场景：编辑修改汽车类型相关信息	▶ ▼
1. 当 点击"编辑"按钮修改"奔驰"价格"500000",再修改回"540000"	>
2. 那么 验证是否修改成功	>

图 10 场景 6

场景 7：进入动态页面，并新增动态“今天喜提宝马，开着很爽”

7. 场景：新增动态相关	▶ ▼
1. 当 跳转到动态网站 "http://localhost:9876/news"	>
2. 那么 点击"新增"按钮，输入标题"今天喜提宝马，开着很爽"，点击"确定"按钮	>

图 11 场景 7

场景 8：进入系统管理下的用户管理页面，添加用户信息，如下表：

表格 4 添加用户信息表

用户名	昵称	年龄	性别	地址
11	11	11	男	三峡大学
22	22	22	女	三峡大学



图 12 场景 8

场景 9：跳转到主页查看 echars 图在修改用户数量后的地域比例改变



图 13 场景 9

场景 10：进入一个新页面，跳转到登录页面，并登录普通用户“zhang”



图 14 场景 10

场景 11：普通用户“zhang”与管理员“admin”在聊天室发起线上聊天



图 15 场景 11

4.2.2 场景对应项目运行 UI 截图

场景 1，如下图：

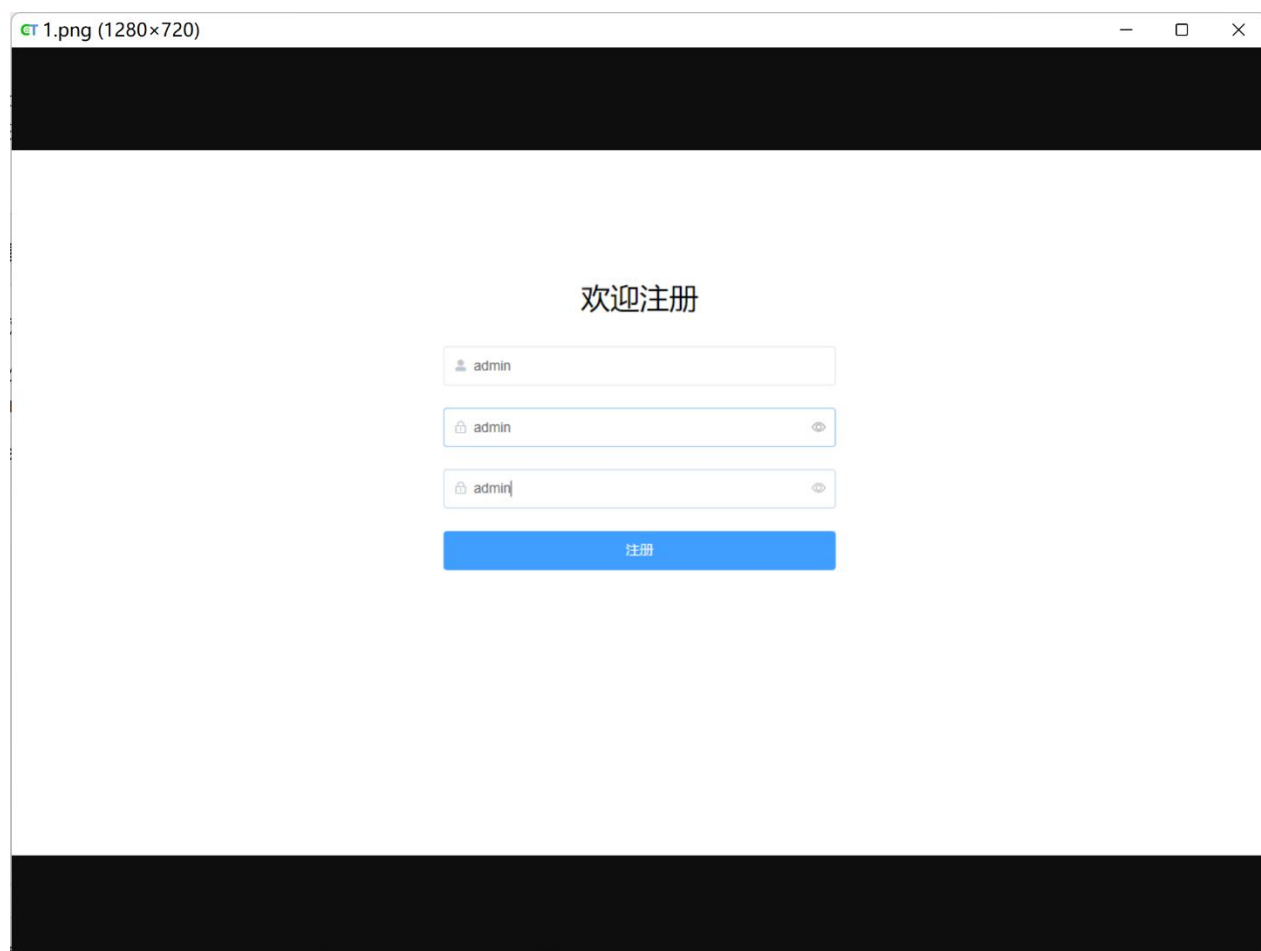


图 16 场景 1

场景 2，如下图：

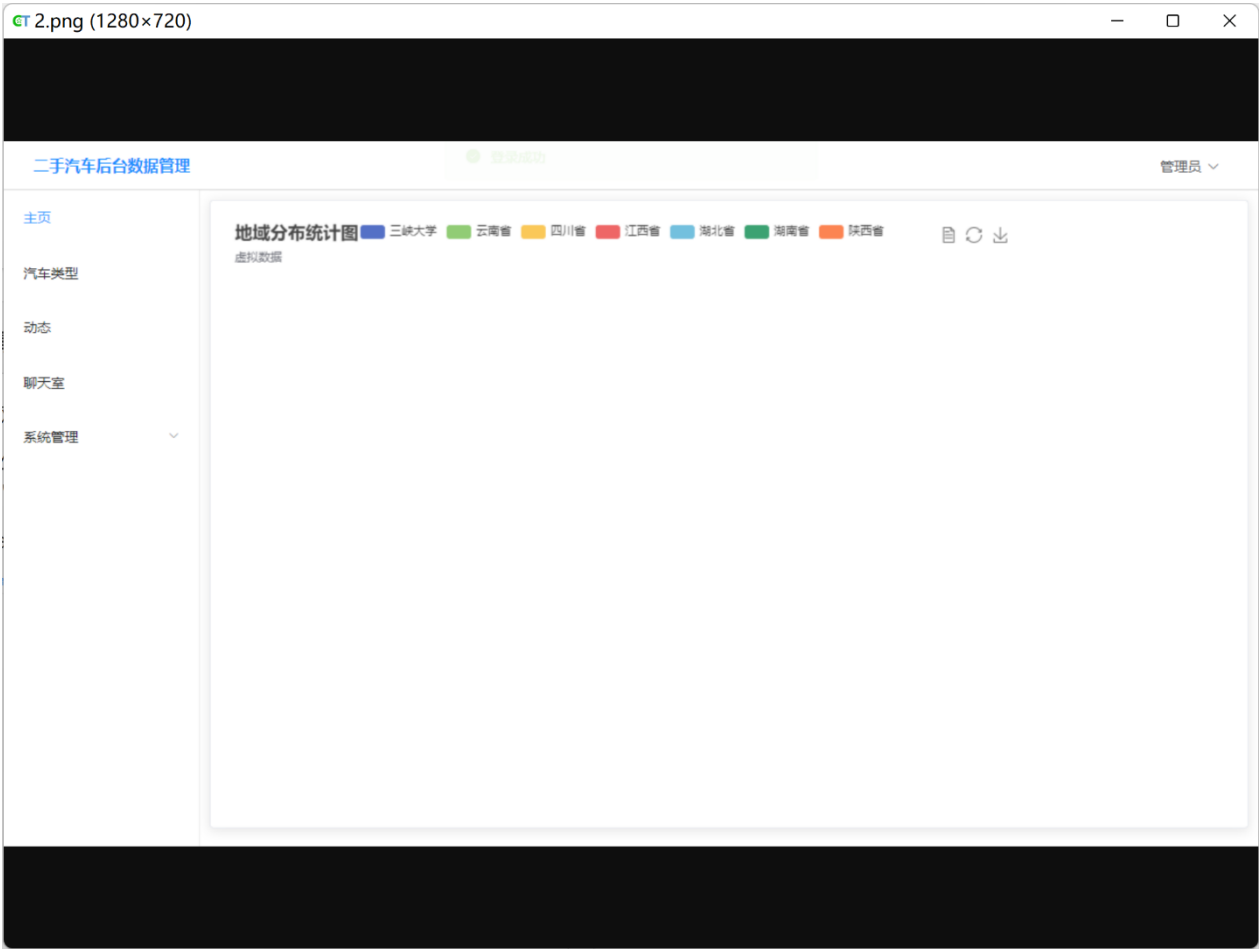


图 17 场景 2

场景 3，如下图：

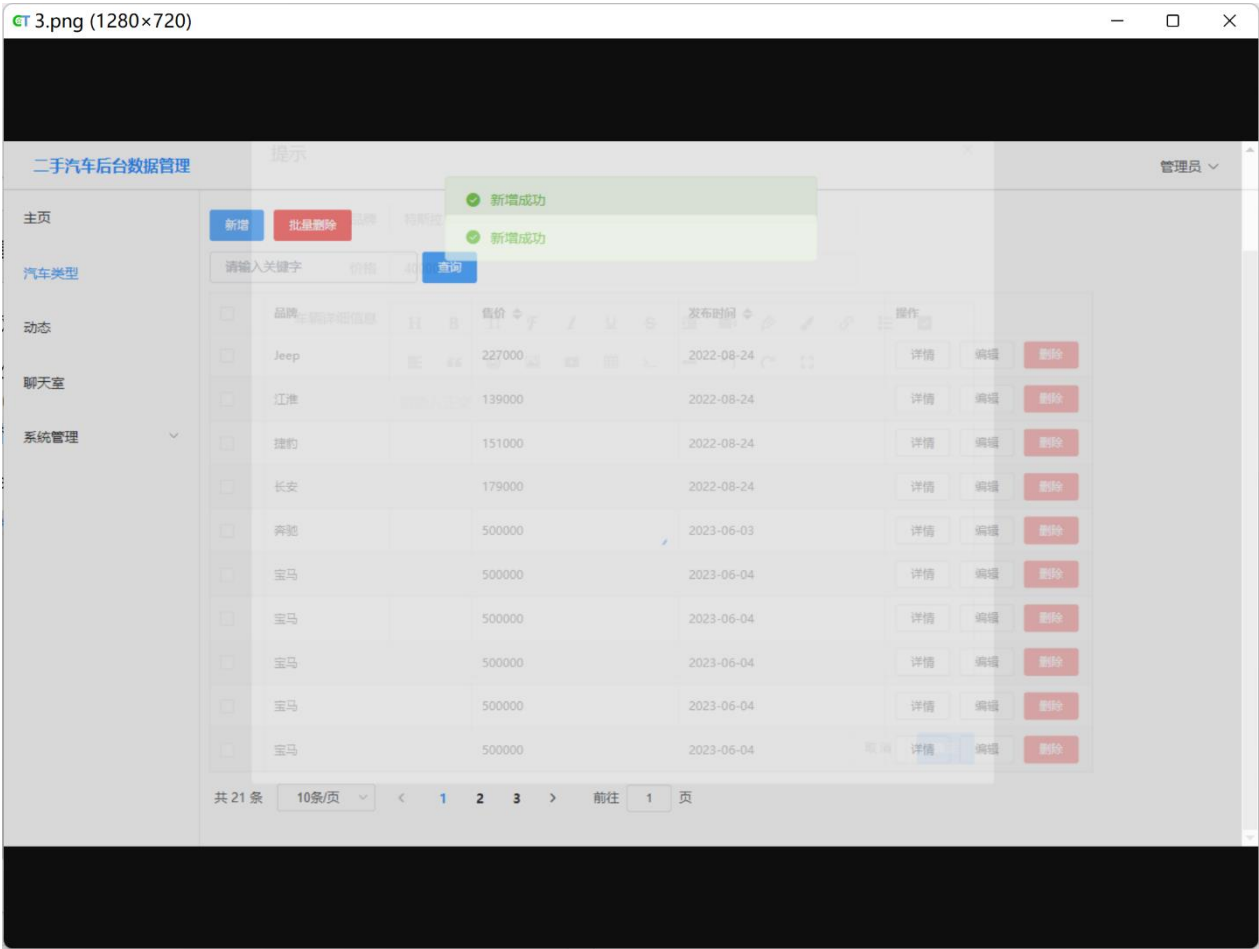


图 18 场景 3

场景 4，如下图：

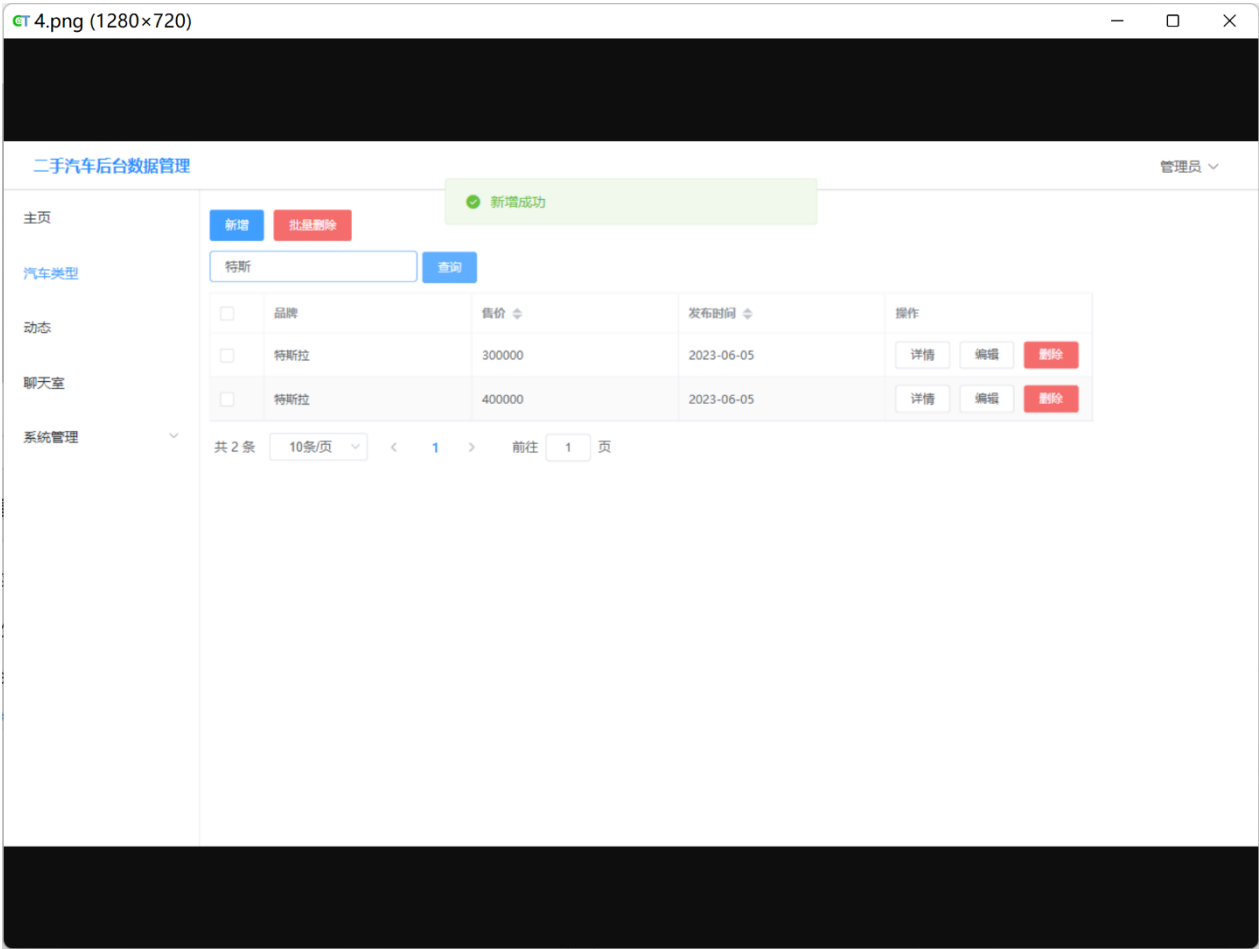


图 19 场景 4

场景 5，如下图：

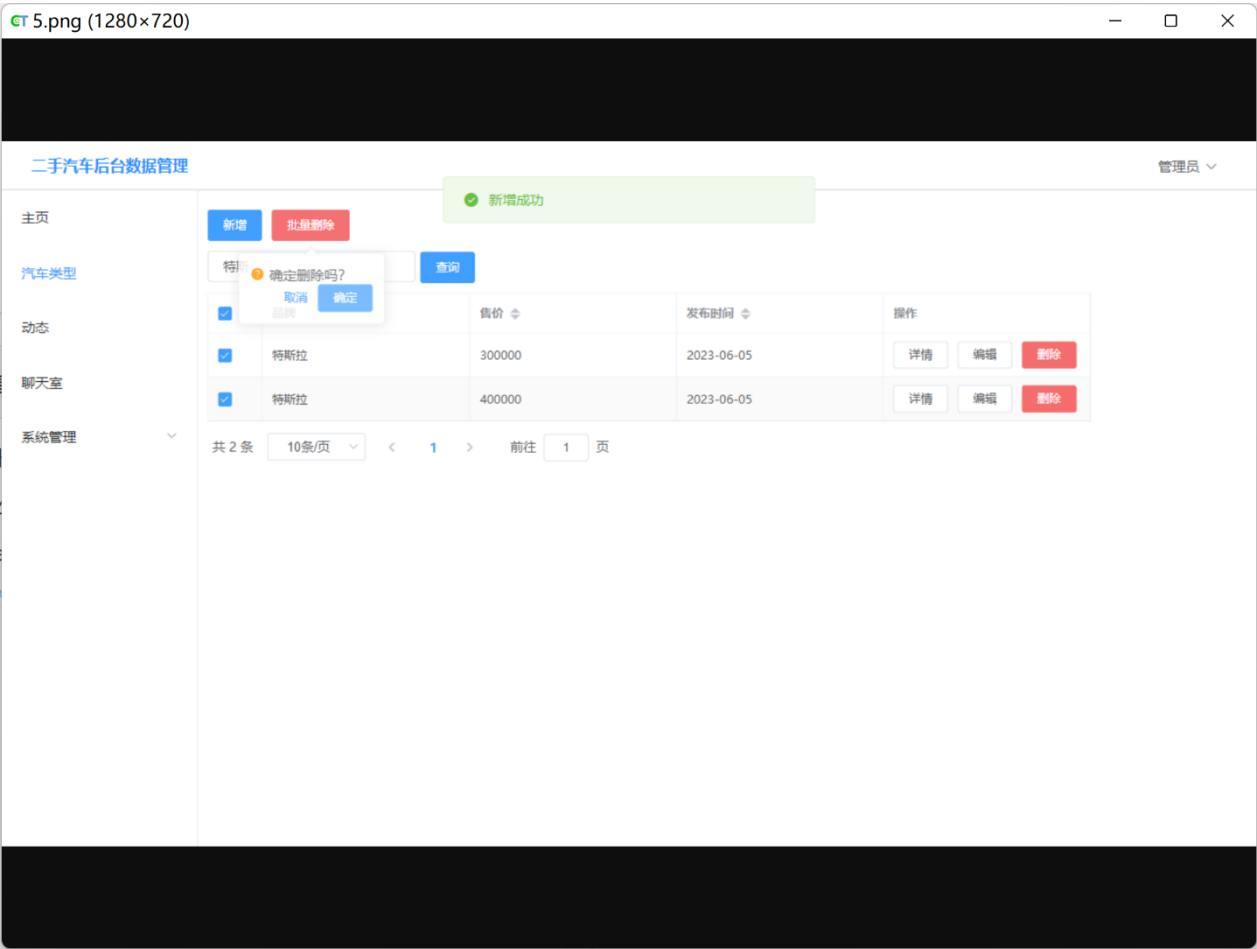


图 20 场景 5

场景 6，如下图：

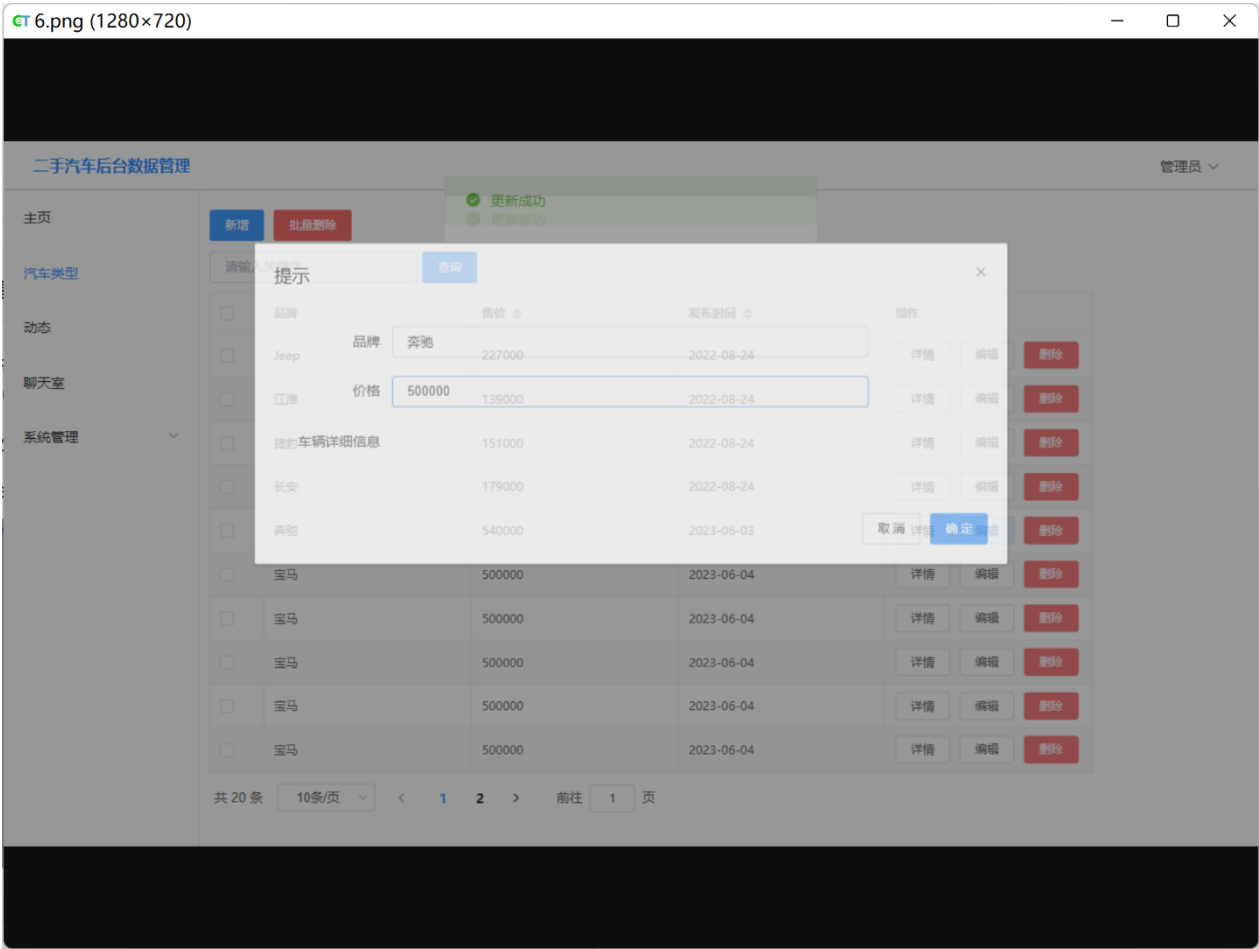


图 21 场景 6

场景 7，如下图：

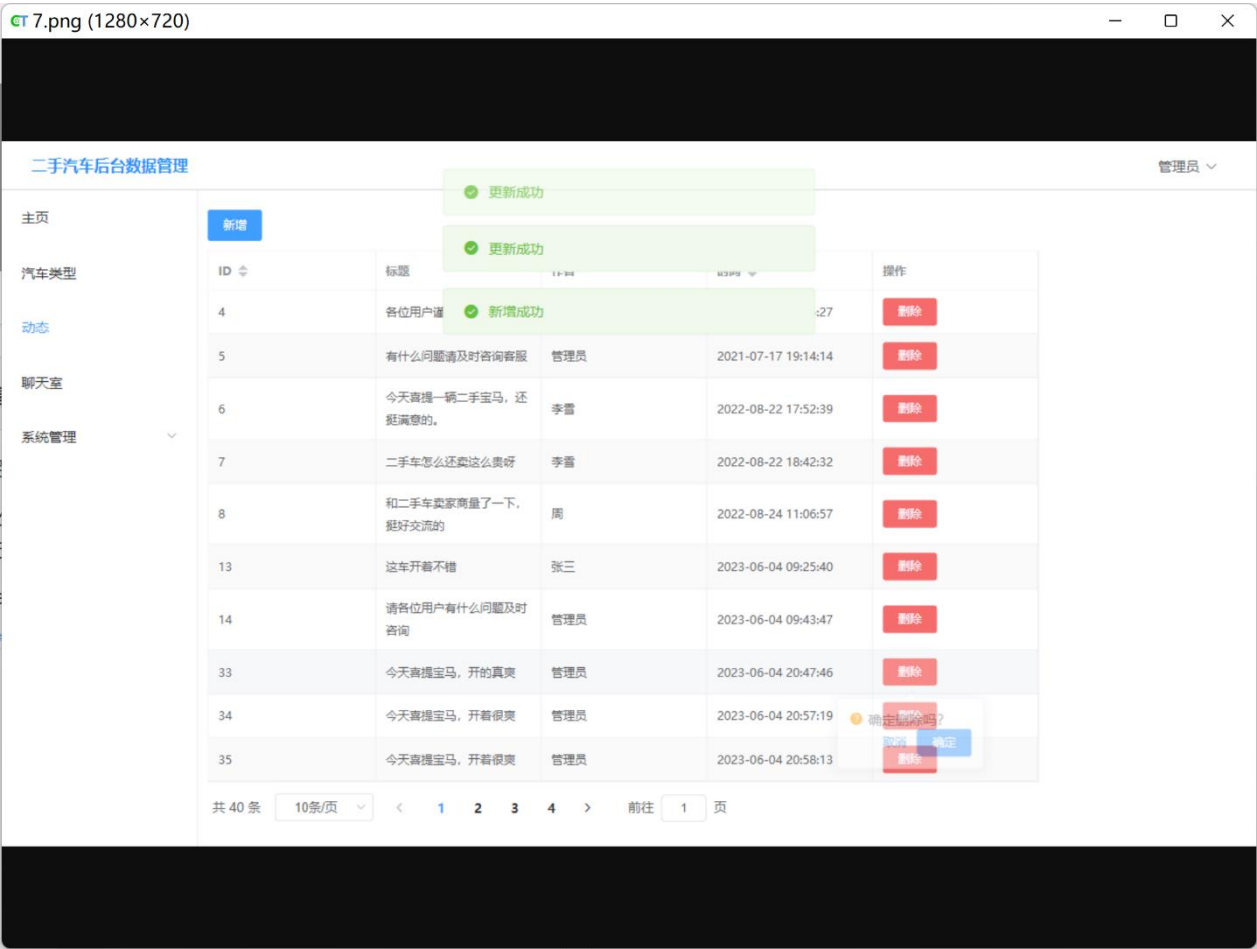


图 22 场景 7

场景 8，如下图：

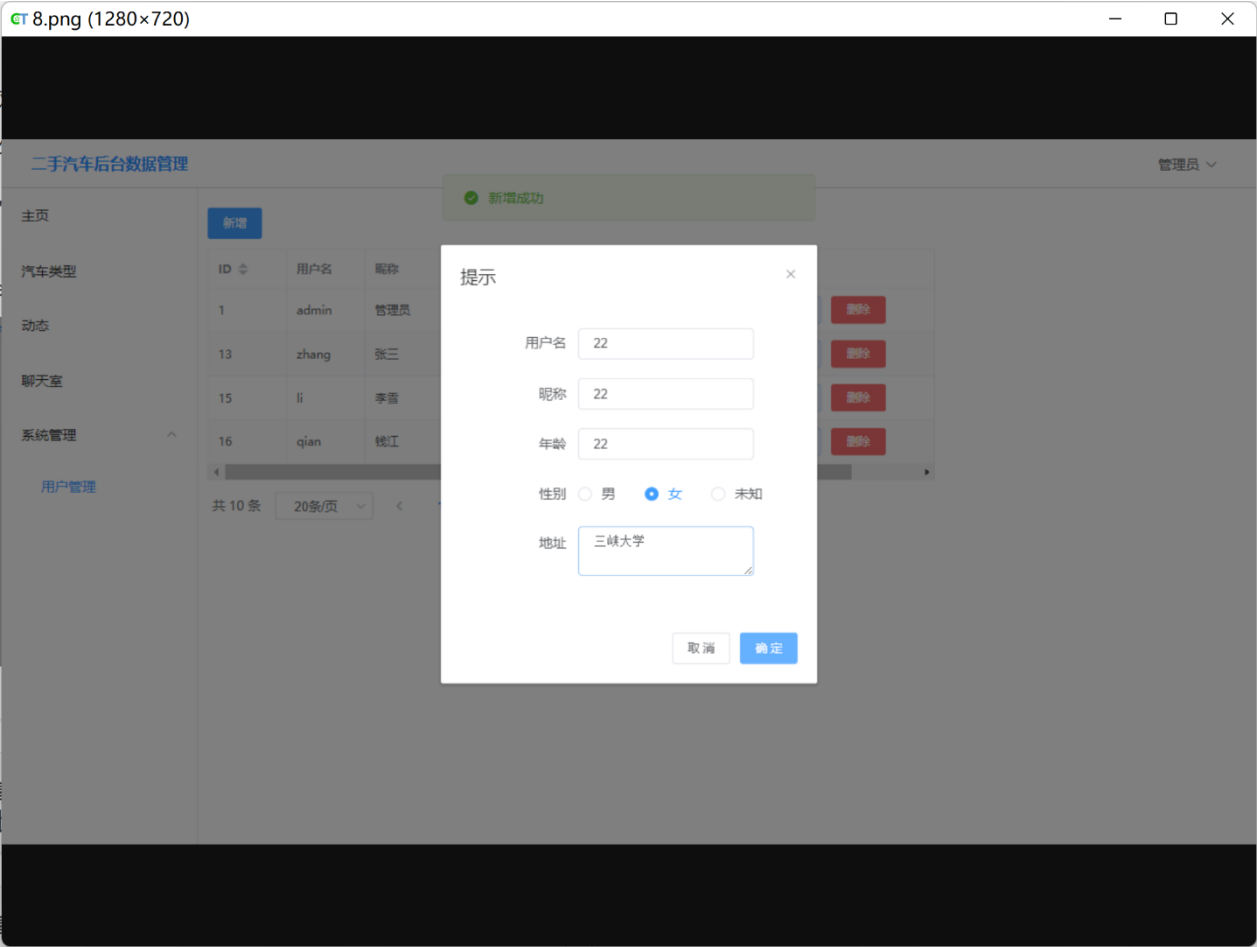


图 23 场景 8

场景 9，如下图：

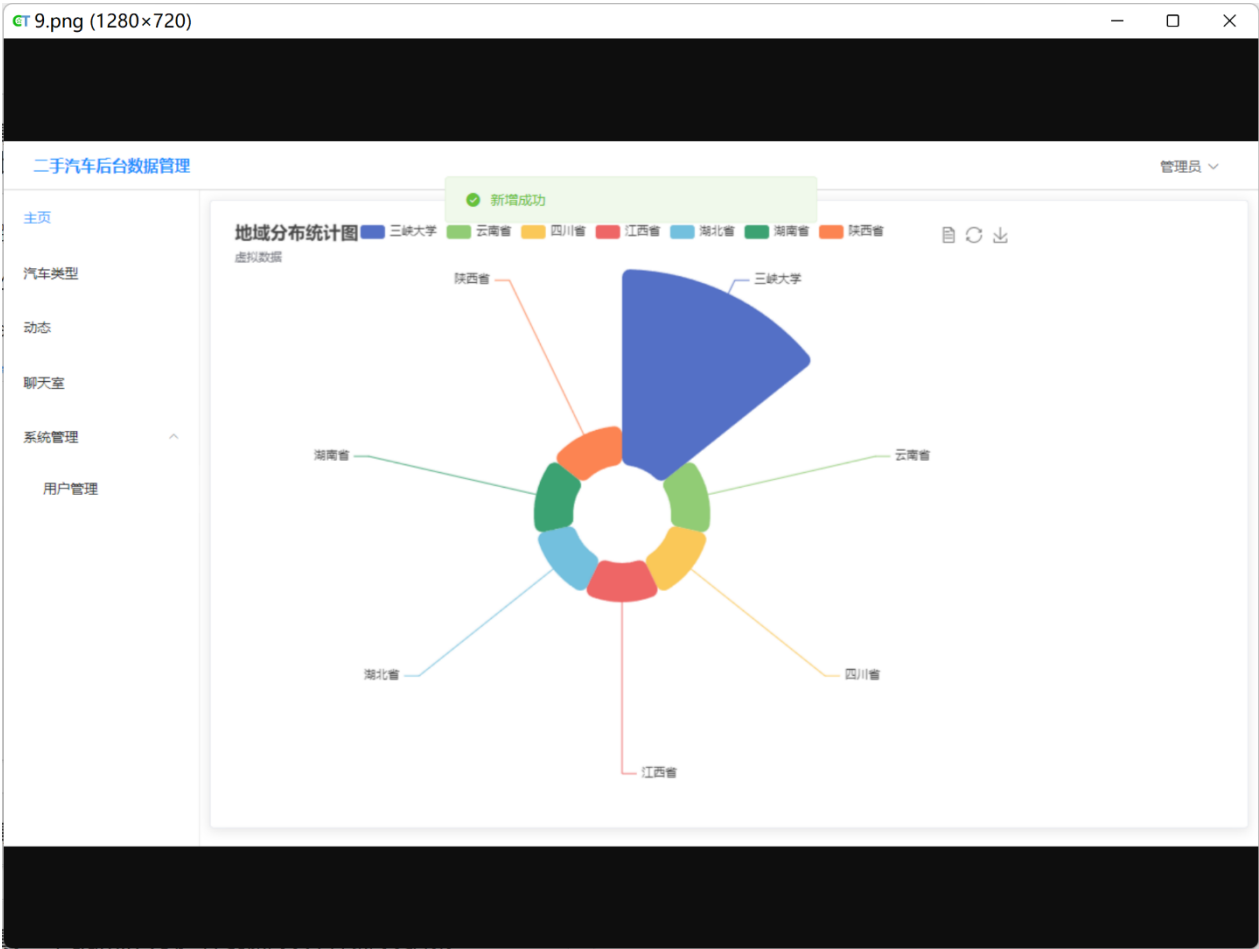


图 24 场景 4

场景 10，如下图：

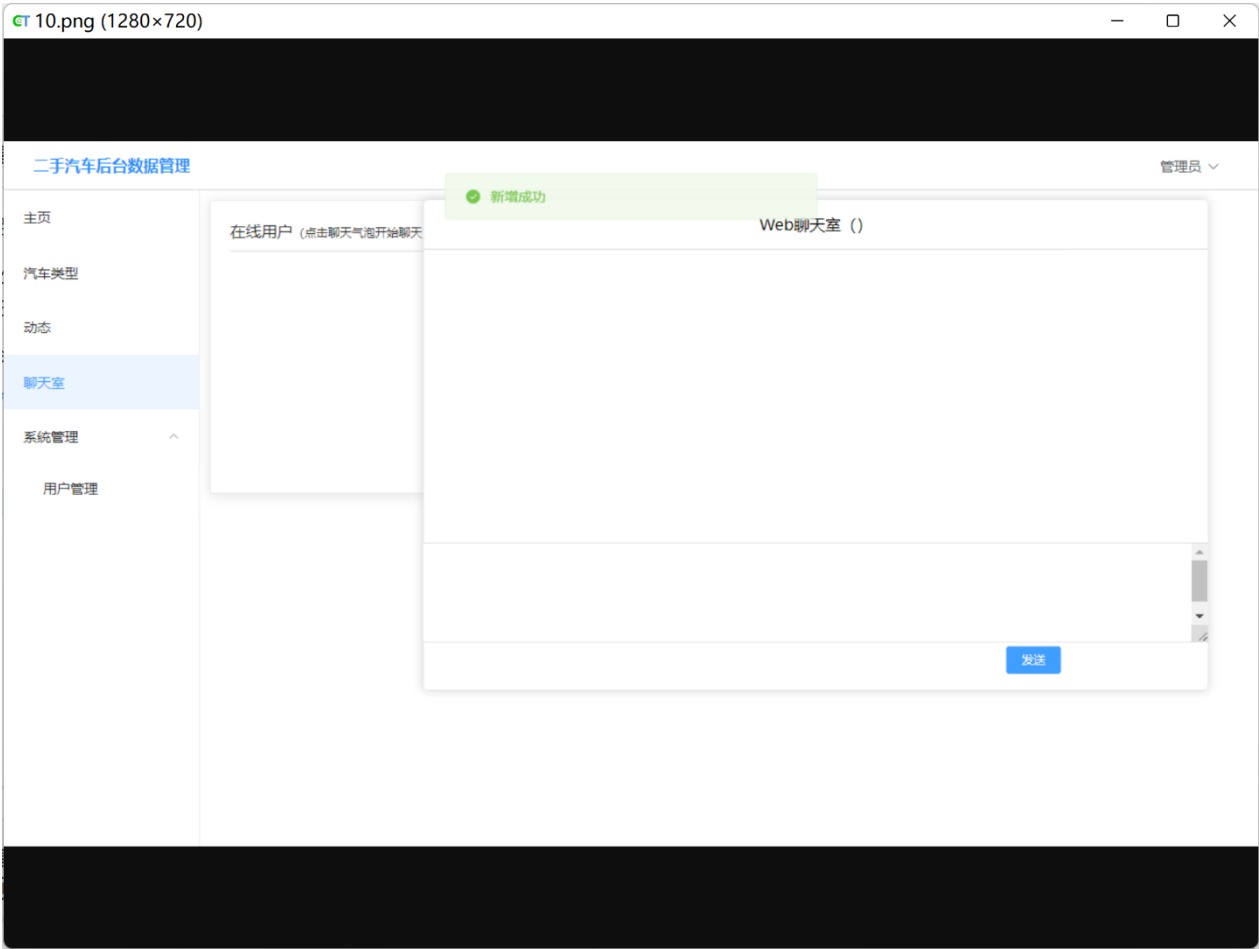


图 25 场景 10

场景 11，如下图：

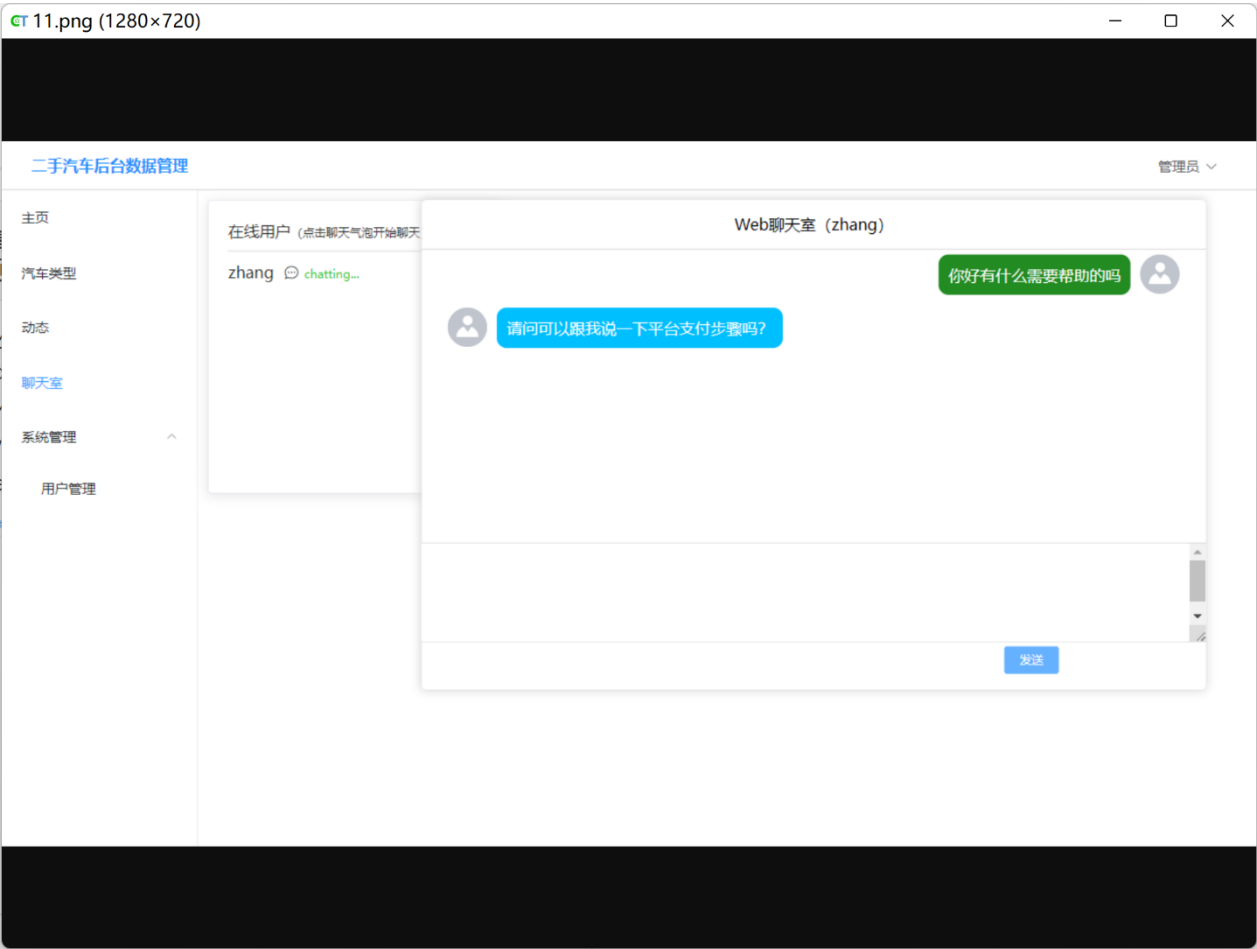


图 26 场景 11

4.2.3 测试报告及通过率

测试报告如下图：

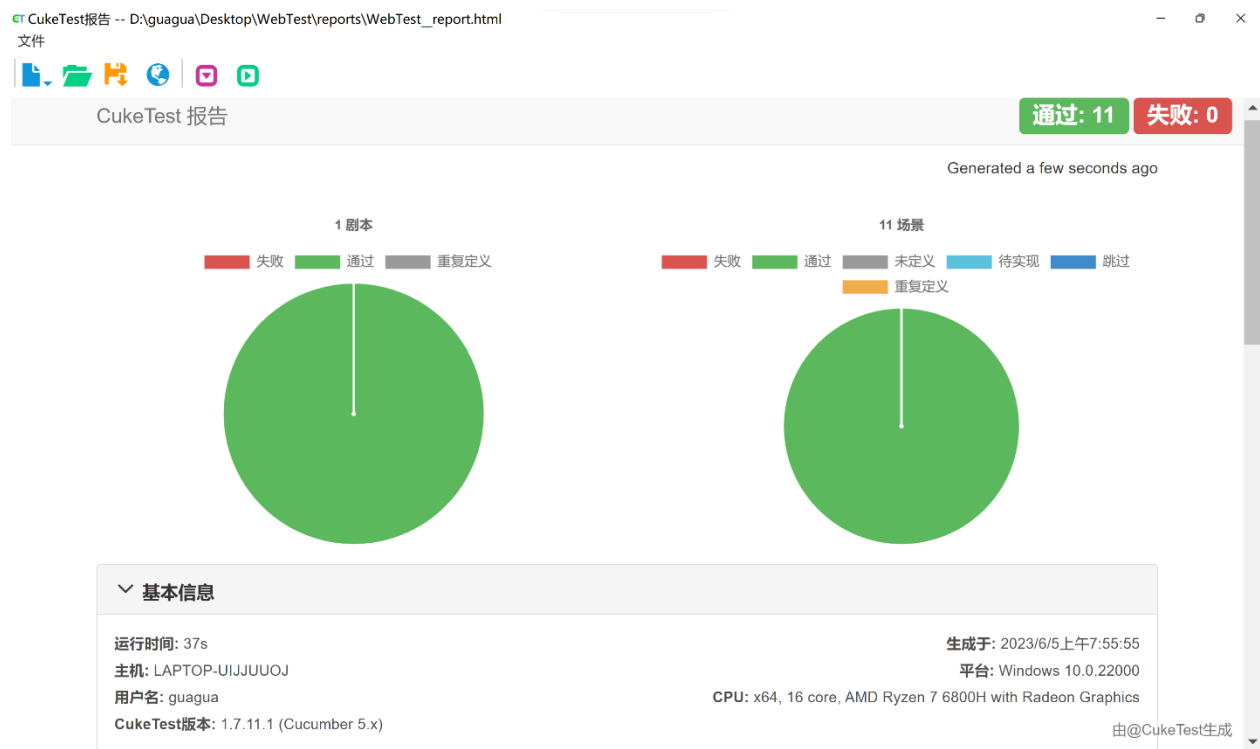


图 27 测试报告

测试通过率为：100%，如下图：

CukeTest报告 -- D:\guagua\Desktop\WebTest\reports\WebTest_report.html	
文件	
功能: <二手汽车后台数据自动录入并检验相关功能>	
登录注册admin账户，并输入汽车类型，动态，用户信息等数据，更改后查看主页echars图的变化，并与在线用户zhang进行聊天	
场景: 注册admin账户	通过
场景: 进入二手汽车后台数据管理系统	通过
场景: 新增汽车类型相关信息	通过
场景: 模糊查询汽车类型相关信息	通过
场景: 批量删除汽车类型相关信息	通过
场景: 编辑修改汽车类型相关信息	通过
场景: 新增动态相关	通过
场景: 新增用户相关信息	通过
场景: 到主页查看echars图在修改用户数量后的地域比例改变	通过
场景: 登录普通用户"zhang"	通过

图 28 测试通过情况

5.结果分析

5.1 整体UI自动化测试结果分析

这次通过 CukeTest 框架一共测试了 11 个场景，这 11 个场景可以看到，在测试报告中都成功通过了。因此本次的 UI 自动化测试还是比较圆满的。

但是其实在编写具体的剧本以及步骤文件时，我还是发现了我项目中的一些小问题，比如说在输入汽车类型以及动态信息时，对于右侧表格中的删除按钮，我在写步骤文件的时候点击了删除按钮以后出现了确定还是取消删除的弹窗，但是无法点击确定按钮，这个是我没有解决的一个问题，我后续查了一些相关资料，有说是代码问题的，也有说是等待时长问题页面为加载成功，还有选择器没有选择成功的问题。我这几种方法都尝试了一下，但是仍就没有解决，因此我把这一部分的内容在我的步骤文件里面删除了，以保证整体项目 UI 测试的一个通过率为 100%，不过这也说明我在这一部分代码书写当中可能存在问题，这也是这个测试框架为我测试出来的一些 bug，虽然这个功能我自己在运行使用的情况并没有出现问题，但是能从测试框架给我的结果中，反映出一些潜在的 bug。

5.2 对Cuketest框架使用感受的总结：

整体还是比较便捷，符合 UI 自动化测试的理念，尤其是剧本的编写可使用中文以及官方教程的易理解度很高，但存在一些小问题，针对一些细节问题如录制页面比例，鼠标滚动失灵，选择器的重复 copy 等，增加了额外的修改时间，作为一个比较“新”的框架，在遇到问题时想要在网上找到人交流解决也比较困难，总之期待 Cuketest 框架的发展。

5.2.1 CukeTest 的优点

其实从我一开始接触到 Cuketest 框架之前，我先接触到的是 BDD 行为测试是从 Cucumber 框架开始的，但是当我想了解 BDD 行为测试的时候，我的搜索引擎首先出现的推荐就是 Cuketest 框架，这吸引了我的注意，我点击进入它的官方文档去了解这个框架是讲什么的，其实可以从我目前下载的这个版本，以及官方发表的教程来看，这个框架应该是一个比较“新”的框架，新框架的好处就是相比较老框架来说他使用的技术更新也更符合当今社会相关软件测试人员的一些需求，就比如我看他的官方介绍中有 QT 自动化测试，自动化 API 和 Web 自动化等集成测试的功能，同时它也继承了 Cucumber 框架以及 JavaScript，可视化编辑调试功能等等，甚至可以跨越多平台应用各种类型的自动化。这也是它为什

么一开始吸引我的原因，实际使用下来，它不需要像 Cucumber 框架一样去 idea 内部引入依赖，只需要简单的将软件下载下来即可，同时 Web 的录制功能也大大减少了我的工作量。同时因为 CukeTest 的官方文档和教材都是中文版的也更有利于进行学习。

5.2.2 CukeTest 的缺点

但也正是因为它的“新”，在面对一些具体功能的时候，它可能就不那么适应。比如说它的页面适应问题，在我录制过程中，无论我原本的项目在谷歌浏览器上是怎么样的适配，但是当我点击录制按钮，它都会变的比例诡异，甚至一些我原本的 UI 界面的按钮和交互功能，没有办法点击到。



图 29 录制网页比例问题

关于交互问题有的时候在录制的时候鼠标会出现滚动无法实现的问题，像这种情况的话就只能重新录一遍，我在录制的时候，其实花费最多精力的不是写剧本文件，是在调整录制比例这种细节的问题，尽管我有寻找录制设置时，增加录制参数 `viewport-size` 但是仍旧没有达到很好的效果。

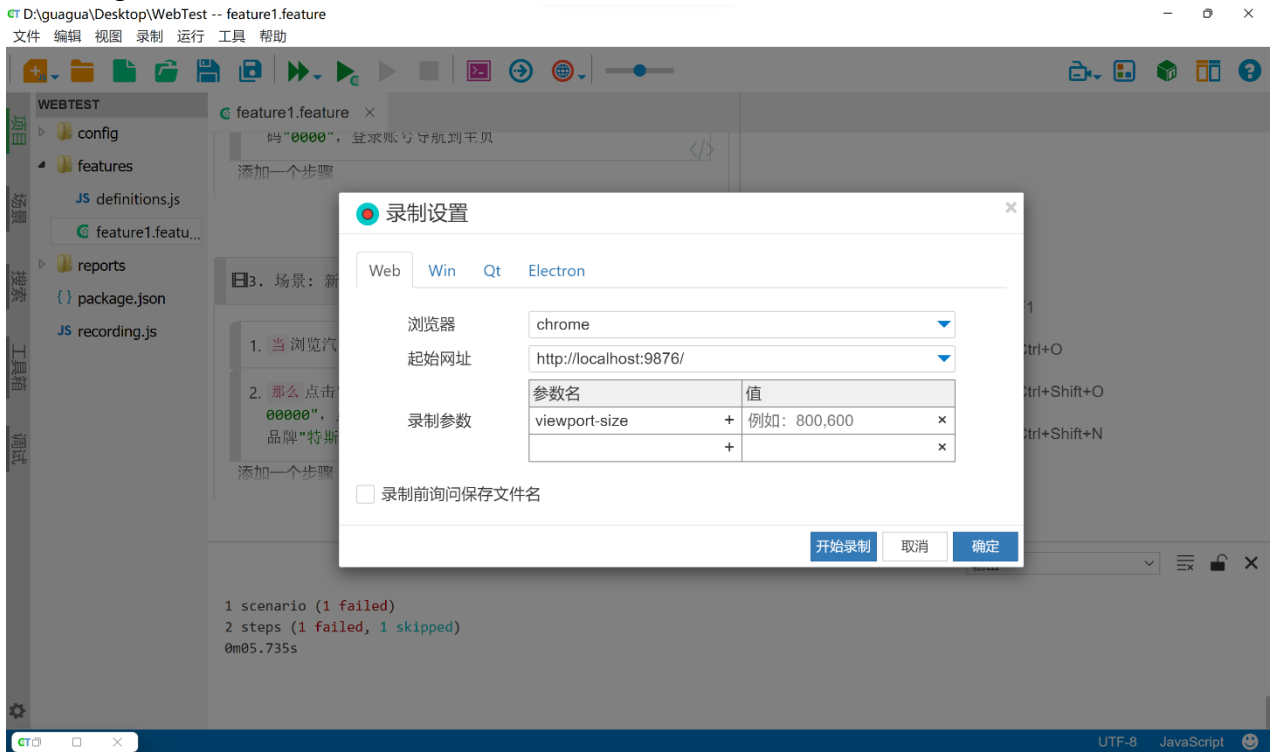
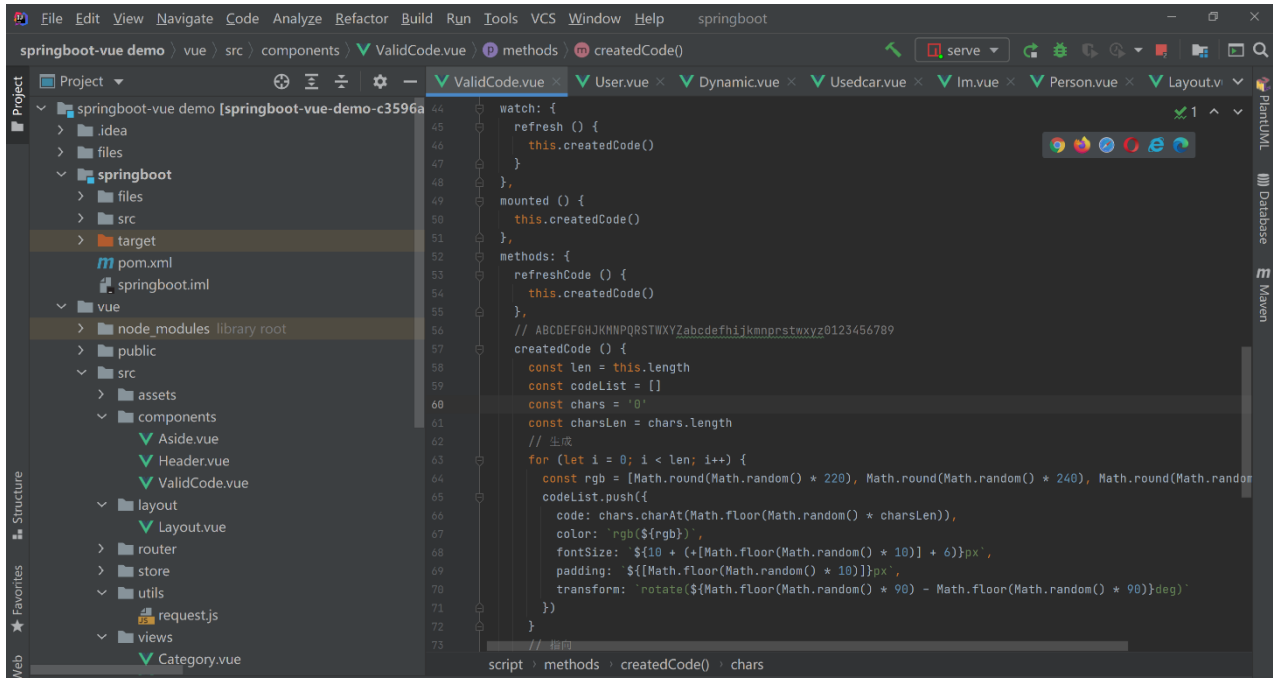


图 30 录制参数设置问题

我觉得这作为一个很新的框架，也是官方需要后续在更新版本时解决的问题，同时在面对一些特定功能，比如说我书写的验证码功能，我原本的验证码是通过一系列字符随机排列组合成 4 个来传递的，一开始录制的时候，它的参数是写死的，虽然后续可以进行参数化调整，但是我的验证码再写了相关随机函数的情况下，仍旧没有办法达到与项目中的验证码一致的结果，因此我最后采取的措施就是将我的验证码字符改为了只留下 0 这个个数字，以保证项目进行的连贯性。



```
44 watch: {
45   refresh () {
46     this.createdCode()
47   },
48 },
49 mounted () {
50   this.createdCode()
51 },
52 methods: {
53   refreshCode () {
54     this.createdCode()
55   },
56   // ABCDEFGHJKMNPQRSTWXYZabcdefghijklmnopqrstuvwxyz0123456789
57   createdCode () {
58     const len = this.length
59     const codeList = []
60     const chars = '0'
61     const charsLen = chars.length
62     // 生成
63     for (let i = 0; i < len; i++) {
64       const rgb = [Math.round(Math.random() * 220), Math.round(Math.random() * 240), Math.round(Math.random() * 255)]
65       codeList.push({
66         code: chars.charAt(Math.floor(Math.random() * charsLen)),
67         color: `rgb(${rgb})`,
68         fontSize: `${10 + ([Math.floor(Math.random() * 10)] + 6)}px`,
69         padding: `${[Math.floor(Math.random() * 10)]}px`,
70         transform: `rotate(${Math.floor(Math.random() * 90) - Math.floor(Math.random() * 90)}deg)`
71       })
72     }
73     // 返回
74     return codeList
```

图 31 验证码后端修改

还有就是关于选择器的问题，这个框架似乎对于复选框一类的选择器不太敏感，几乎每一个有关于复选框的 `recording.js` 我都做了修改，需要重新去页面 copy 选择器，甚至当涉及日期时间这种与时俱进的内容时，我还需要通过时间的变化，不停的 copy 选择器，希望官方能在下次版本的更新中优化这个问题。

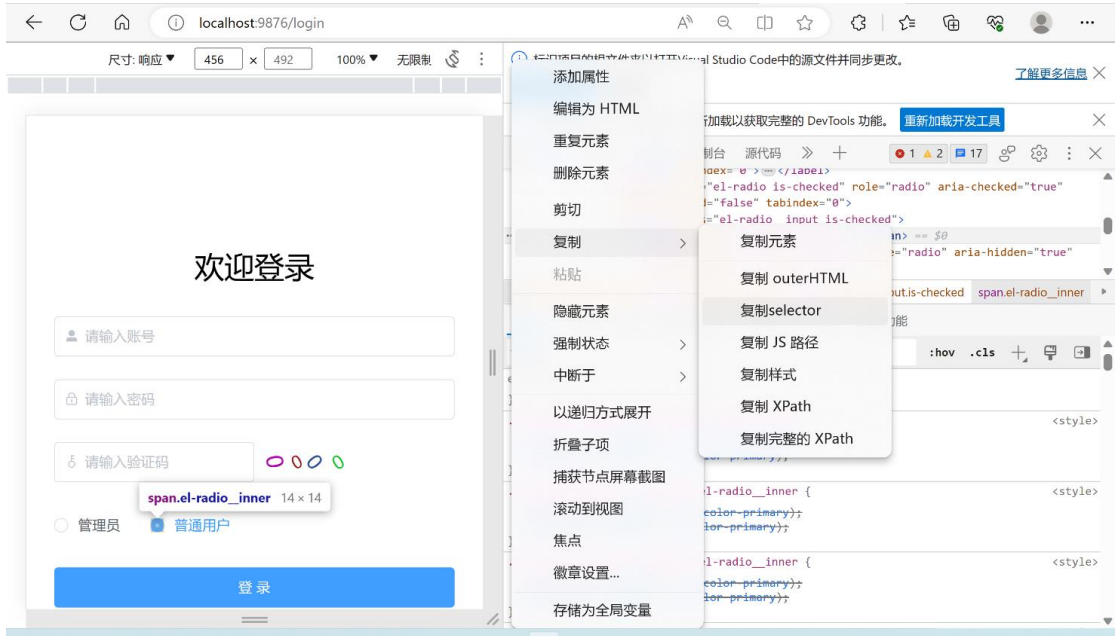


图 32 选择器复选问题

参考文献

- [1] <https://www.bookstack.cn/read/cuketest-2021/>
- [2] <https://github.com/cuketest/demos/tree/master/javascript-zh/web>
- [3] https://www.bilibili.com/video/BV1UD4y1P7bF/?spm_id_from=333.999.0.0
- [4] <https://www.cuketest.com/>
- [5] <https://juejin.cn/post/6844903727980085262>
- [6] <https://blog.csdn.net/aorubi0512/article/details/101790274>
- [7] <https://stackoverflow.com/search?q=CukeTest>
- [8] <https://star.jmhui.com.cn/p1/645.html>
- [9] <https://www.liuchengtu.com/>