**广州大学松田学院**

**题 目**

**基于微服务 SpringCloud的**

**书籍管理系统**

**二〇二一 年 五月**

**基于微服务springcloud的书籍管理系统**

摘要：随着计算机科学技术的快速发展，计算机在各个应用领域已经逐渐普及，并且普遍得到应用。计算机信息的时代，微服务的书籍管理系统管理相当重要，针对这一趋势，我们组设计的微服务书籍管理系统非常实用。该系统利用前后端知识和数据库技术，前端采用Layui+jQuery,后端应用Springcloud+mysql技术实现功能的呈现，采用tomcat服务器实现。

本系统包括eureka服务注册中心、配置中心、提供者user,menu,order、服务网关Zuul、链路追踪zipkin,等一系列管理功能。使用方便，操作简单，安全性能好，功能齐全，借助c/S模式，可以很方便实现登录，实现了数据库的同步共享。

**关键词：springcloud+mysql** IDE工具 Layui+Jquery

目 录

[第1章 绪 论 1](#_Toc30987)

[1.1 系统开发的编写目的、背景 1](#_Toc24171)

[1.1.1 编写目的 1](#_Toc21551)

[1.1.2 系统开发的背景 1](#_Toc30502)

[1.2参考资料 2](#_Toc24492)

[第2章 系统需求分析 2](#_Toc22345)

[2.1 可行性分析 2](#_Toc11409)

[2.1.1 经济可行性 2](#_Toc18763)

[2.1.2 技术可行性 3](#_Toc27290)

[2.1.3 操作可行性 3](#_Toc30891)

[2.2 系统的需求分析 3](#_Toc12076)

[2.3 系统的功能需求 4](#_Toc1400)

[2.3.1 注册服务中心模块 4](#_Toc349)

[2.3.2 配置中心模块： 4](#_Toc3268)

[2.3.3 提供者模块： 4](#_Toc15097)

[2.3.4 网关zuul管理模块： 5](#_Toc5647)

[2.3.5 链路追踪管理模块： 5](#_Toc5647)

[第3章 系统概述 5](#_Toc3805)

[3.1 用户登录 5](#_Toc5296)

[3.2 查询书籍管理 7](#_Toc18223)

[3.3 添加书籍管理 7](#_Toc26042)

[3.4 处理订单管理 8](#_Toc8752)

[3.5 用户管理 9](#_Toc13821)

[3.6 查询用户 9](#_Toc10938)

[3.7 添加用户 10](#_Toc22794)

[第4章 项目设计 12](#_Toc4529)

[4.1 系统设计 12](#_Toc8553)

[4.1.1 数据库表的主要实现 13](#_Toc21586)

[4.2 项目模块介绍 16](#_Toc22253)

[4.2.1 服务注册服务器模块eureka-server 16](#_Toc16487)

[4.2.2 配置中心模块configserver 18](#_Toc21255)

[4.2.3 提供者模块 19](#_Toc1331)

[4.2.4 网关模块gateway-zuul 2](#_Toc27387)6

[4.2.5 请求链路追踪模块zipkin-server 2](#_Toc21868)9

[4.3 系统的开发平台及运行环境 3](#_Toc20649)2

[4.3.1系统的开发平台 3](#_Toc17779)2

[4.3.2系统的运行环境 3](#_Toc12549)2

[第5章 系统性能 3](#_Toc22580)3

[5.1 性能需求 3](#_Toc3556)3

[5.2 安全性需求 3](#_Toc28742)3

[5.3软件质量属性 3](#_Toc28478)4

[第6章 系统测试 3](#_Toc22580)5

[6.1 系统测试 3](#_Toc3556)5

[6.1.1 测试的意义 3](#_Toc3556)5

[6.1.2 测试的目的 3](#_Toc3556)5

[6.1.3 软件测试方法 3](#_Toc3556)5

# **第1章** 绪 论

## **1.1 系统开发的编写目的、背景**

### **1.1.1 编写目的**

21世纪是知识经济的时代，是人才竞争的时代，随着时代的发展，人类已进入信息社会，信息产业正成为全球经济的主导产业。信息自动化的作用也越来越大，特别是各行各业的管理领域，智能化信息处理已经是提高效率、规范管理、客观审查的最有效途径。

微服务书籍管理系统是一个不可或缺的订书管理系统，它的内容对于消费者和店家来说都是至关重要的，他对于用户的买书过程有着非常重要的作用。

### **1.1.2 系统开发的背景**

随着计算机的普及和信息技术的发展，人们的生活发生了日新月异的变化，各类计算机软件逐渐渗透到了社会的每个角落，大大地改善了人们的生活质量，提高了人们的工作效率。而微服务书籍管理系统就是对用户买书的取得、开发、保持和利用等方面进行计划组织、指挥和控制活动。研究人与人的关系，调动人的积极性，实现组织的既定目标，使经济效益最大化。

微服务书籍管理系统主要提供书籍订购和查询用户管理整体解决方案，实现认识操作的合理化和电子化，下在提高工作效率的同时，将用户和管理员战略发展紧密结合，从而实现系统的最大利用化。

总的来说，微服务书籍管理系统主要提供书籍订购系统的基本功能，包括买书管理、查询书籍管理、添加书籍管理、处理订单管理和用户管理以及各种网关、链路追踪等功能。

## **1.2参考资料**

黑马程序员编著，Java EE 企业级应用开发教程（Spring+Spring MVC+My Batis）,人民邮电出版社

黑马程序员编著，Spring Cloud微服务架构开发教程,人民邮电出版社

陈光剑编著，《Spring Boot实战 开发》，机械工业出版社

# **第2章** 系统需求分析

## **2.1 可行性分析**

### **2.1.1 经济可行性**

从经济上分析，开发一个人事管理系统所需的成本不高，只需要使用电脑就能进行开发。IntelliJ IDEA开发工具功能强大，里面包含各种JAR包的集成，运用JAR导入Java实体类以及控制器加Service，可以快速开发，后台数据库管理采用SQLYog或navicat就能进行数据库的导入。开发过程可以节省时间和大量的人力。前期的开发不需要投入太多的人力，开发周期页不长，可以说系统在经济上有一定的可行性。

### **2.1.2 技术可行性**

系统开发工具选择

开发工具：jdk8+tomcat服务器+mysql+IntelliJ IDEA+maven

数据库设计工具：SQLyog+PowerDesigner

面向对象：需求者

我们选择IntelliJ IDEA做开发工具，这是因为界面设计简单，在代码设计中可写Java实体类以及业务逻辑层。运用SSM框架的开发技术，以及方便快捷、灵活使用的mysql，应用范围广，使用mysql是开发系统的最佳搭档，以此诠释了系统的技术可行性

### **2.1.3 操作可行性**

操作可行性是指所开发系统的界面是否美观、简单易操作，是否有操作步骤的提示，是否能自动生成或更新相关信息。从运行的顺畅度看，可以大概知道系统的可行性和普及性。该微服务管理系统采用的是springcloud、Spring+SpringMVC+MyBatis框架开发技术，在当前属于高端流行的技术。界面简洁美观，用户操作非常简单。

使用本系统，可以为管理员减少工作用量以及费用负担，同时也满足用户买书查询书籍基本信息的检查需求。该系统运行速度迅速，可以被用户接受，保证了系统操的可行性。

## **2.2 系统的需求分析**

微服务书籍管理系统，能够实现智能化的书籍订购管理，还要能提供管理员查询书籍的基本信息，要求系统具备以下特点：

1. 操作简单，易用。
2. 数据存储可靠，具备较高的处理效率。
3. 系统安全、稳定，提高信息处理速度和准确性，建立信息平台。
4. 开发技术先进、功能完备、扩展性强。
5. 提高系统 安全性、可靠性和可控性。

## **2.3 系统的功能需求**

微服务书籍管理系统需求说明：

微服务书籍管理系统包含买书管理、用户管理、注册服务中心管理、服务网关管理、配置中心和链路追踪管理等模块。

### **2.3.1 注册服务中心管理模块**

查询所有注册端口的信息：登录localhost:8761可以显示所有已经注册成功的端口信息

### **2.3.2 配置中心模块：**

存储所有端口的配置文件，所有端口是否能够成功开启都需要经过访问配置中心的配置文件，否则无法开启端口

### **2.3.3 提供者模块：**

1. 书籍信息提供者信息：这个模块主要是可以使用本用例对书籍基本信息进行“查询，增加，修改，删除”等操作
2. 实现dao层业务逻辑，并实现书籍所有信息的分页功能

### **2.3.4 服务网关zuul管理：**

1. 调用client端口信息：开启网关端口，输入localhost:8060会直接跳转到client的端口的相关页面，表明服务网关功能成功实现
2. 实现网关与hystrix结合熔断：断开client端口，开启注册服务端口和网关端口，会出现连接异常的显示信息，表示功能实现

### **2.3.5 链路追踪管理：**

**追踪网关、client服务：开启服务网关zuul端口，使用浏览器访问localhost:8060/login.html会直接跳转到localhost:8030/login.html，再打开相关页面，可以追寻到网关已经调用的接口链路**

# **第3章** 系统概述

## **3.1 用户登录**

登录页面以简明的形式提供给用户一个登录界面。用户登录后会显示当前系统的基本信息。

**用户登录页面**

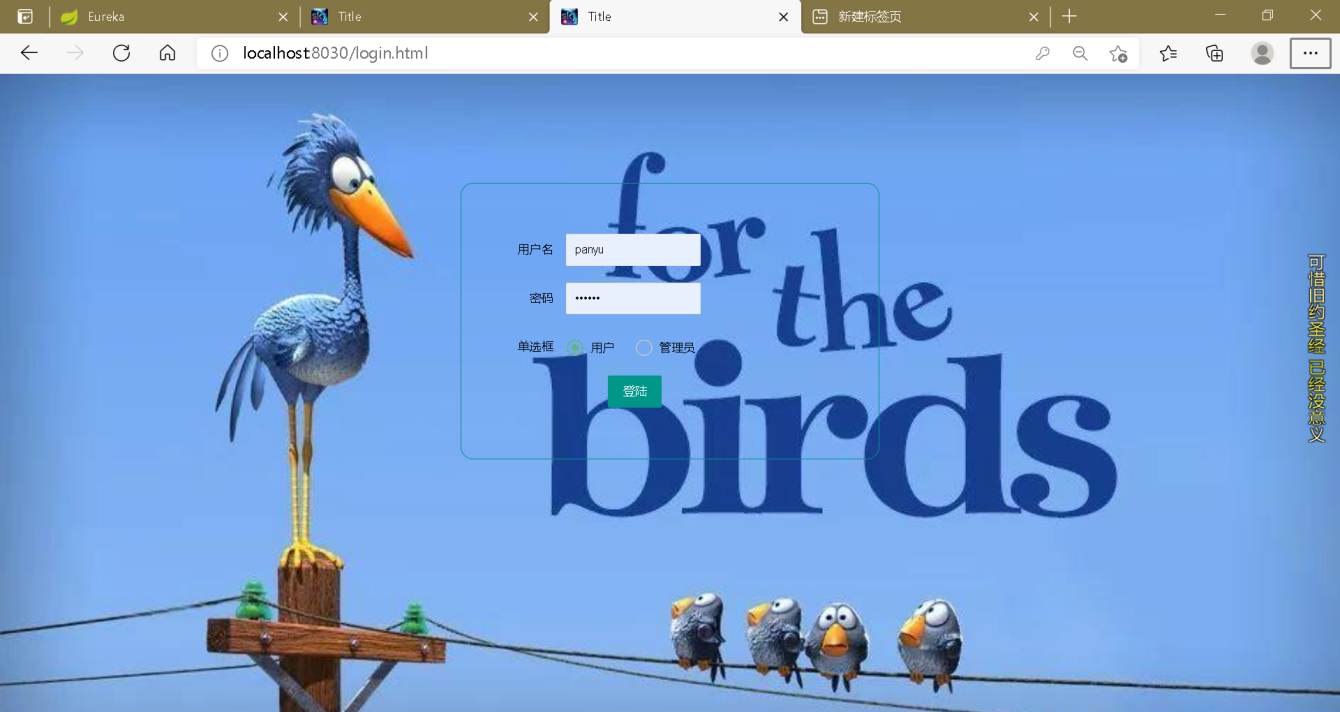


图3.1 用户登录界面图

## **3.2 查询书籍管理**

本系统中的查询书籍管理是一个重要的功能，不可或缺，如下截图可以显示当前后台管理系统中查询书籍信息的页面截图，简单而高效，如下用户信息截图：

**查询书籍管理页面:**

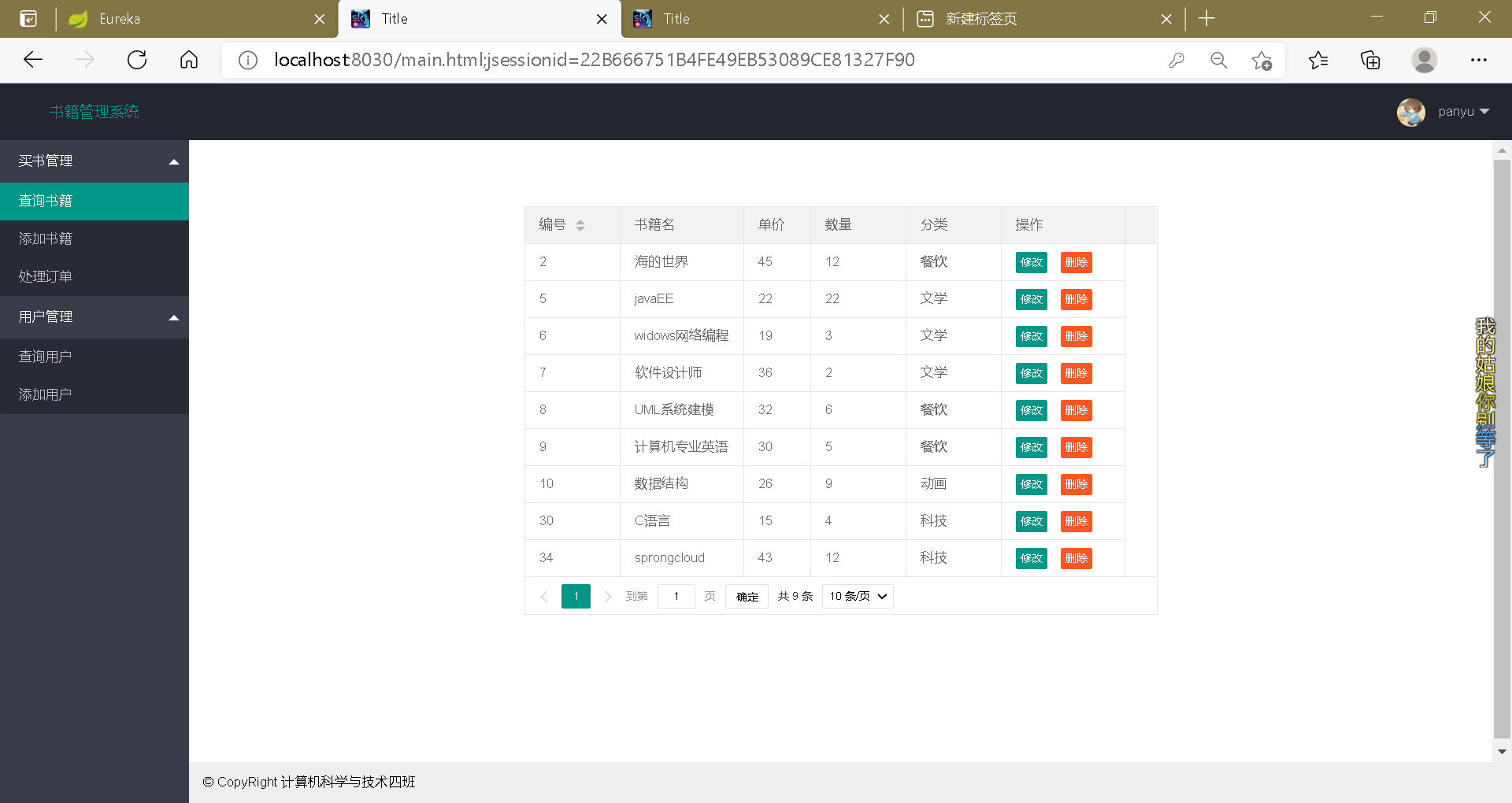


图3.3 查询书籍管理图

## **3.3 添加书籍管理**

添加书籍管理用于添加书籍，使用户可以使用不同角色进行的功能，设计功能有用而高效，如下截图

**查询书籍管理页面:**

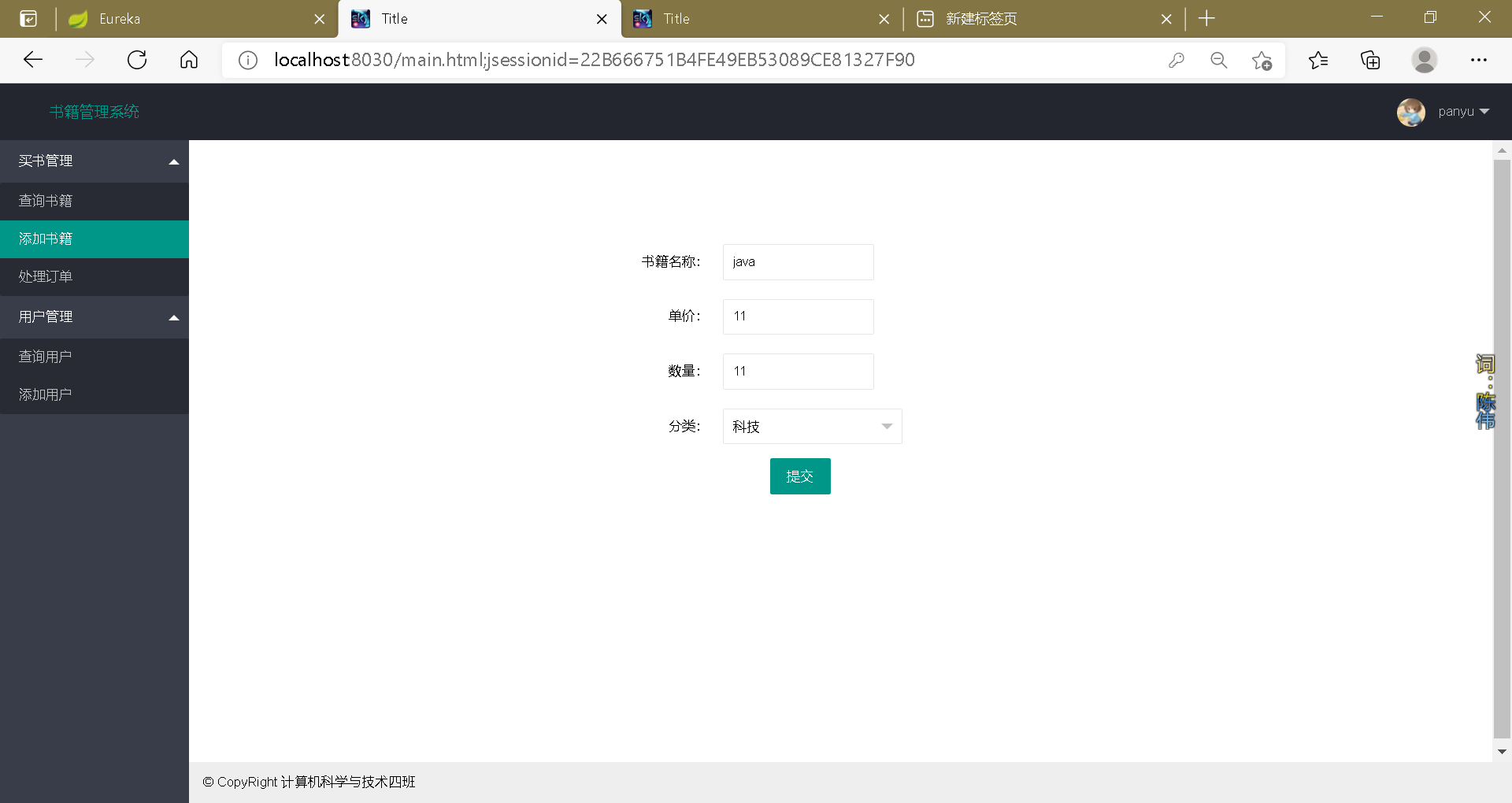


图3.4 添加书籍管理图

## **3.4 处理订单管理**

处理订单管理用于查询书籍是否已经派送，管理员可以进行书籍派送操作，该模块设计功能高效而实用，如下截图

**处理订单管理页面:**

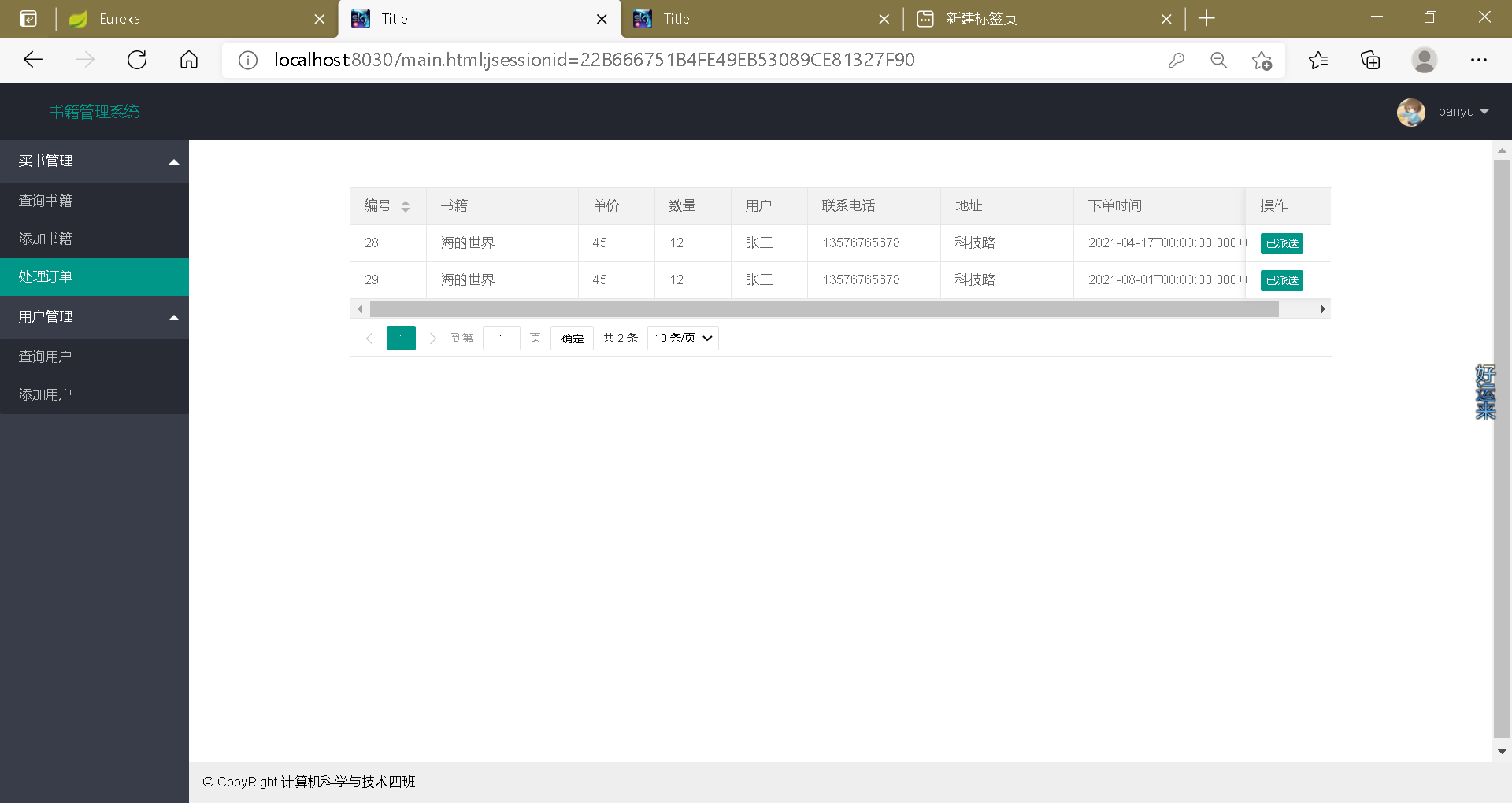


图3.5 处理订单管理图

## **3.5 用户管理**

用户管理用于对用户信息进行管理，管理员可以对用户的信息进行增删查改。

## **3.6 查询用户**

查询用户用于对用户注册信息进行管理，系统管理员可以对用户信息进行增删查改。

**查询用户页面：**

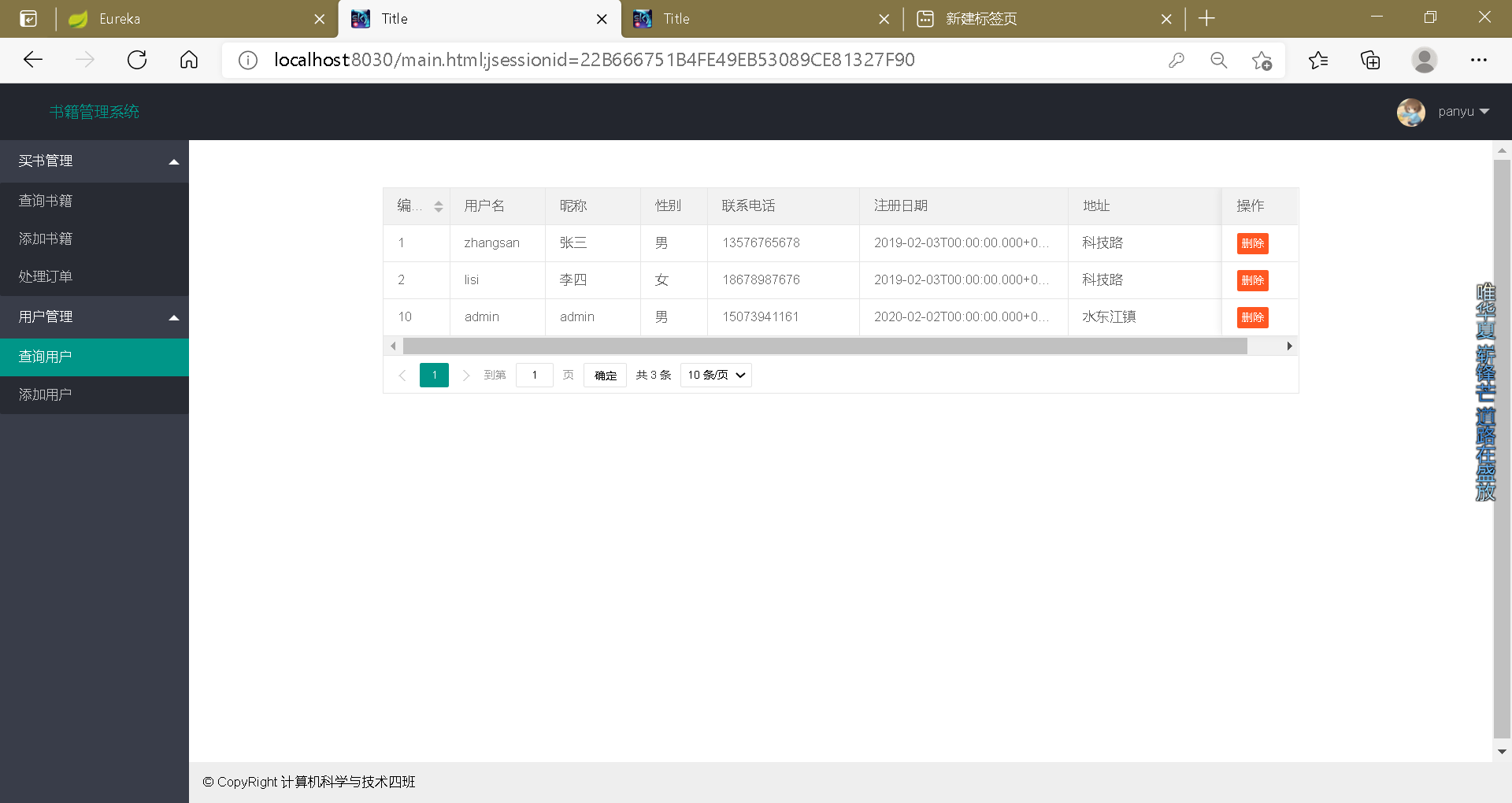


图3.7 查询用户图

## **3.7 添加用户**

添加用户管理用于对用户信息进行添加管理，系统管理员可以进行添加用户

**添加用户页面：**

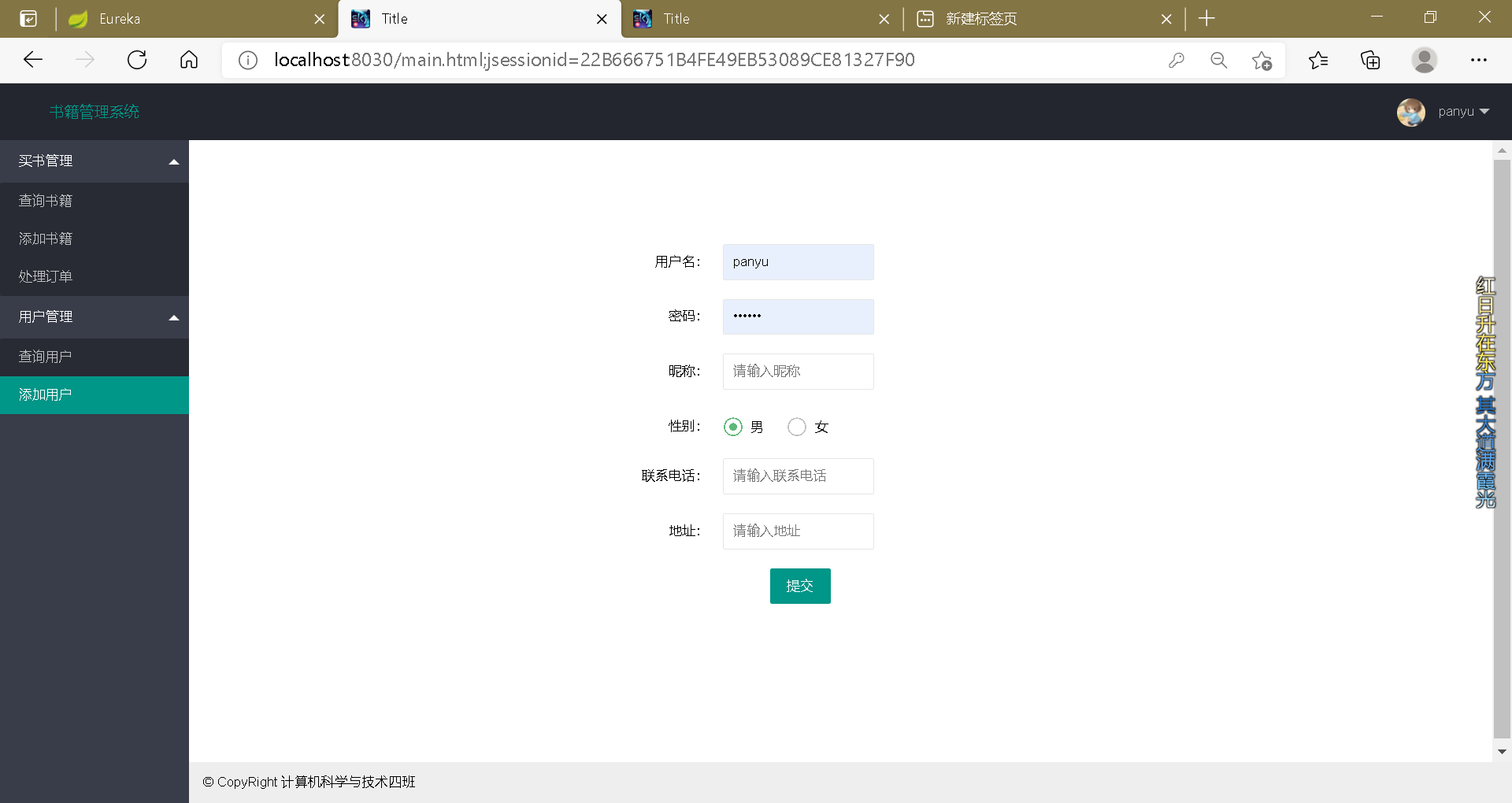


图3.8 添加用户图

# 项目设计

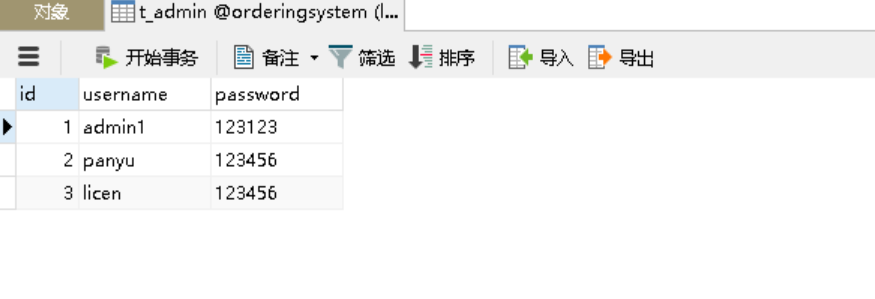
## **4.1 系统设计**

## **4.1.1 数据库表的主要实现**

数据在数据库中地位是数一数二的，应该将其安全性、完整性、可恢复性以及可控制性进行着重考虑，通过数据库对数据实施一定的保护措施，下面将简单介绍部分主要表的设计：

1. t\_admin表，用户登录权限的信息
2. 表3-1 t\_admin表

t\_admin表数据:



1. t\_user表，存储角色资源的信息

表3-2 t\_user表

t\_user表数据：



1. t\_menu表，用户拥有的资源信息

t\_menu表数据:



1. t\_order表，部门信息

3-4 t\_order表

t\_order 表数据：

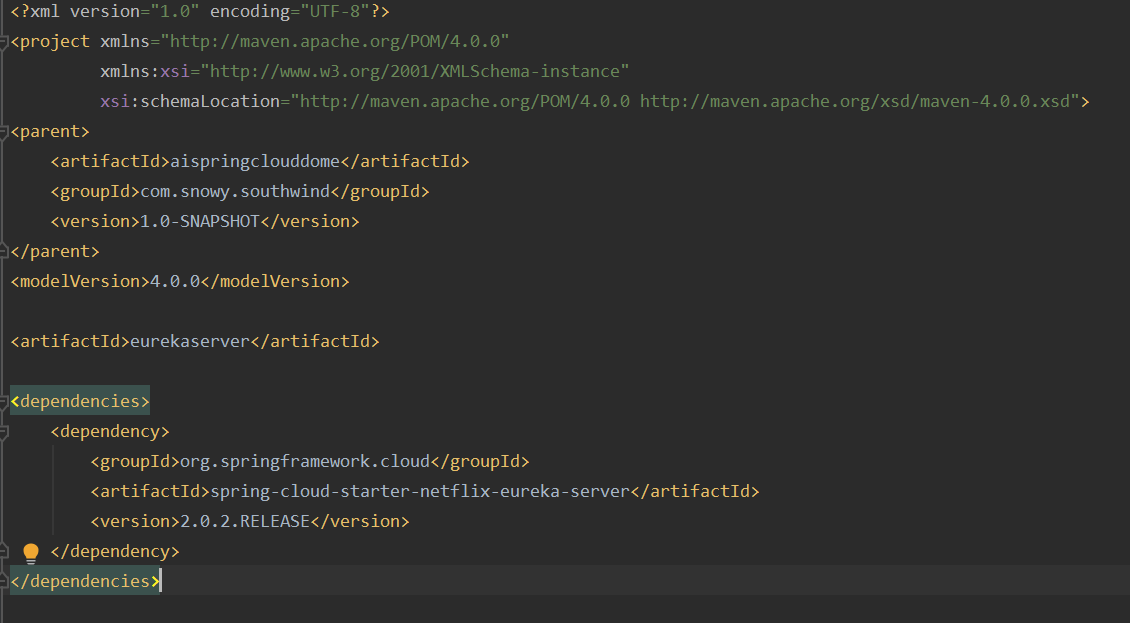


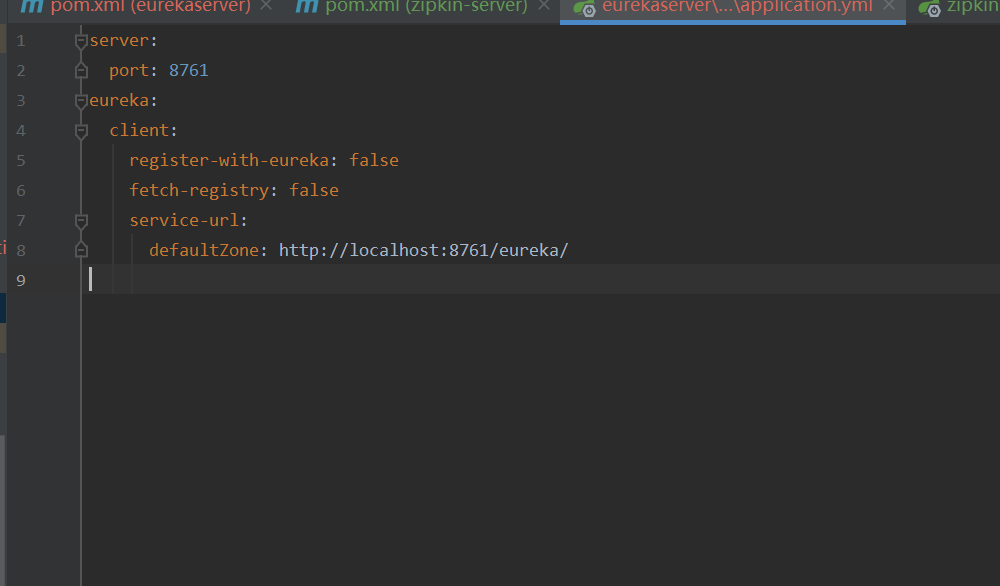
## **4.2 项目模块介绍**

## **4.2.1 服务注册服务器模块eureka-server**

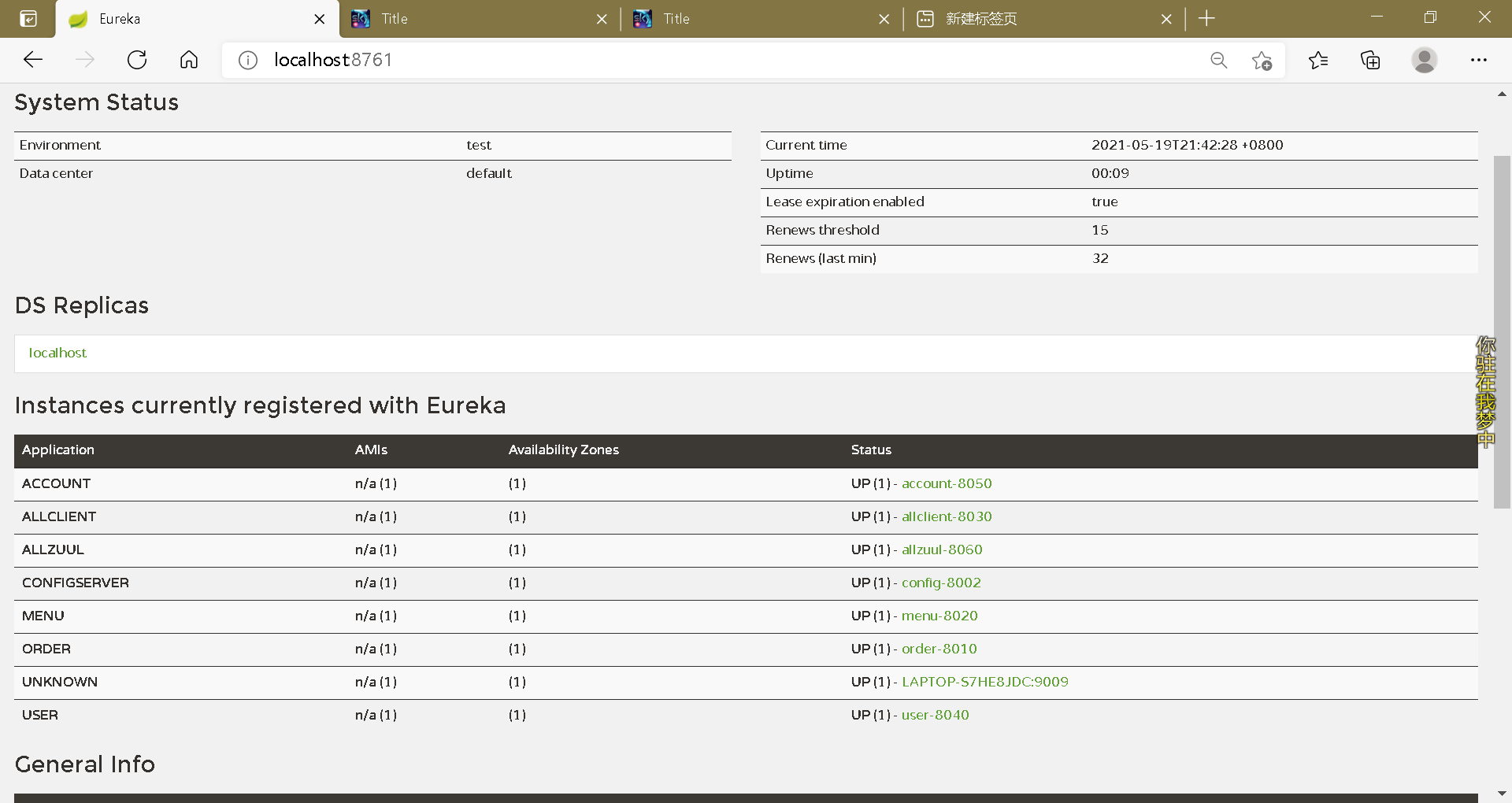
服务注册服务器模块存在的意义使各模块能被注册和识别，几乎所有的Spring Cloud模块功能的实现都离不开服务的发现与注册。本项目使用服务注册服务器模块实现服务注册，配置中心，服务网关zuul,zipkin请求链路追踪。

1. 项目搭建好后的pom.xml文件代码如下所示：

（2）全局配置文件如下：



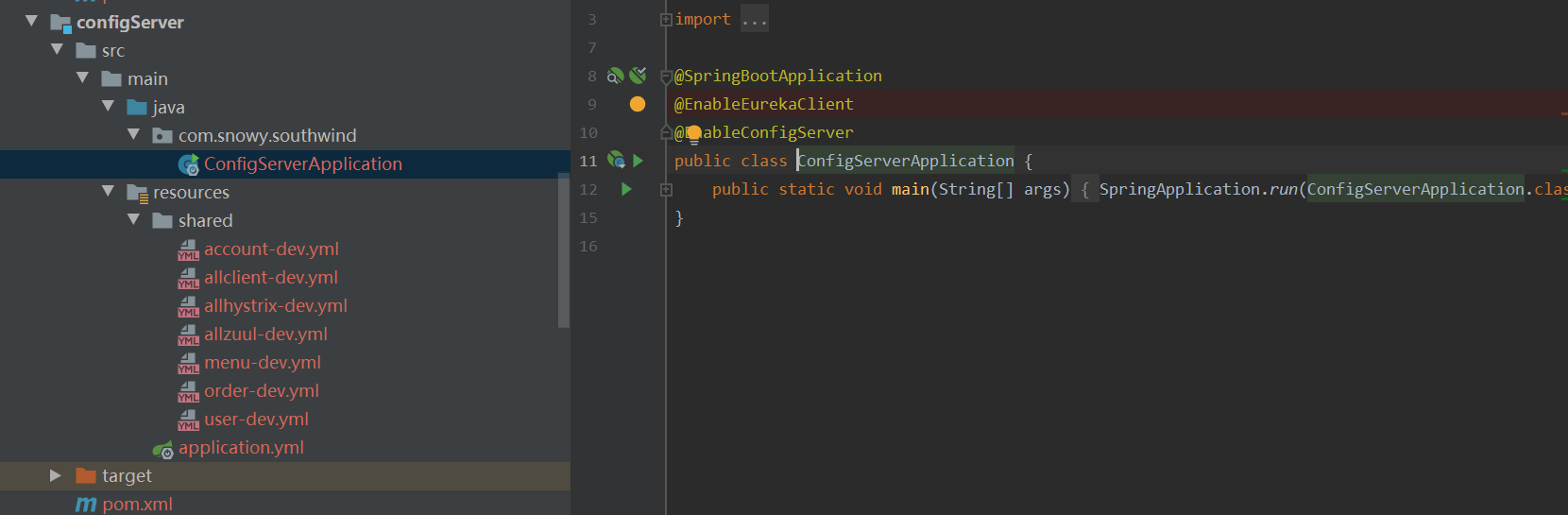
（3）在项目启动类EurekaSeverApplication上添加@EnableEurekaServer注解，开启EurekaSever功能。

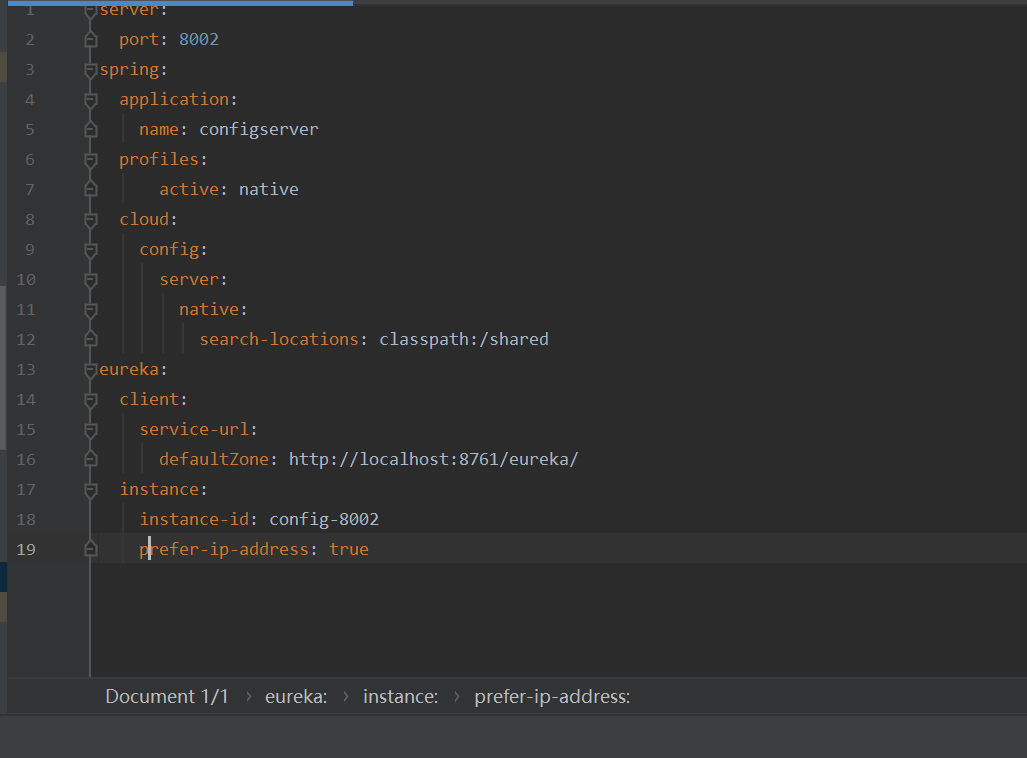
（4）运行eureka-sever。使用浏览器访问**[http://localhost:8761](http://localhost:9020/eureka/),页面如下：**  


## **4.2.2 配置中心configserve**

配置中心存在的意义是存放各个模块的配置文件，提供者menu，order,user,服务网关zuul.链路追踪zipkin的配置文件都在里面。

项目搭建如下所示：

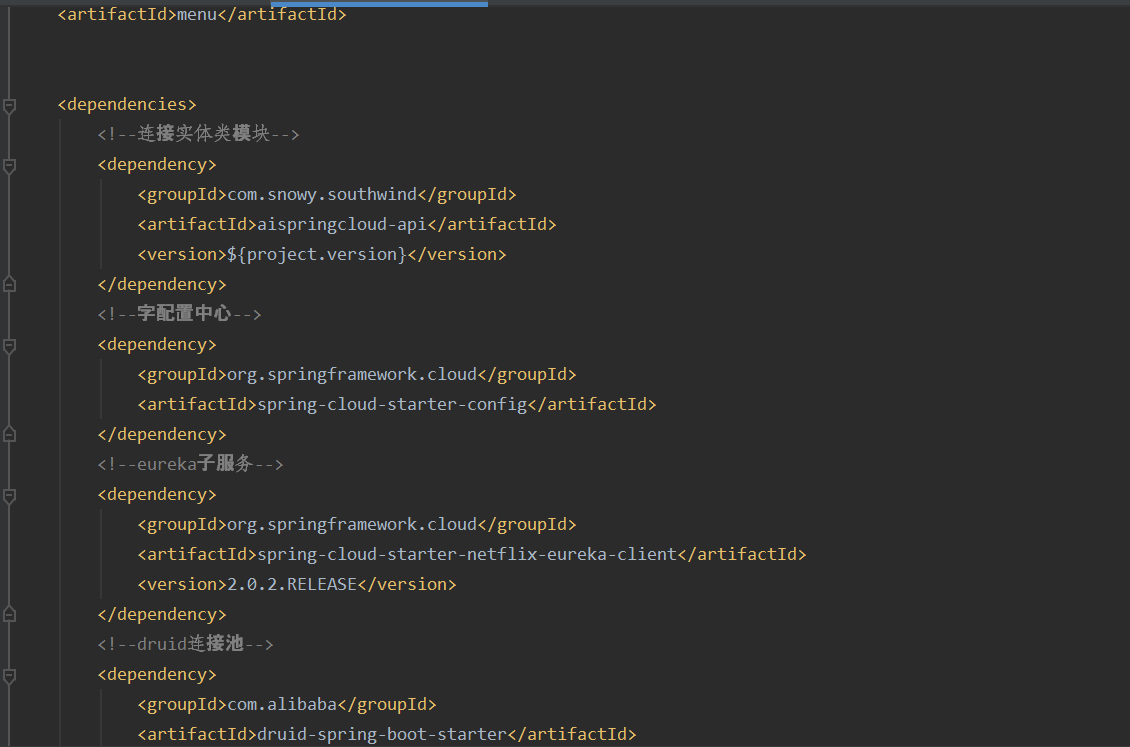


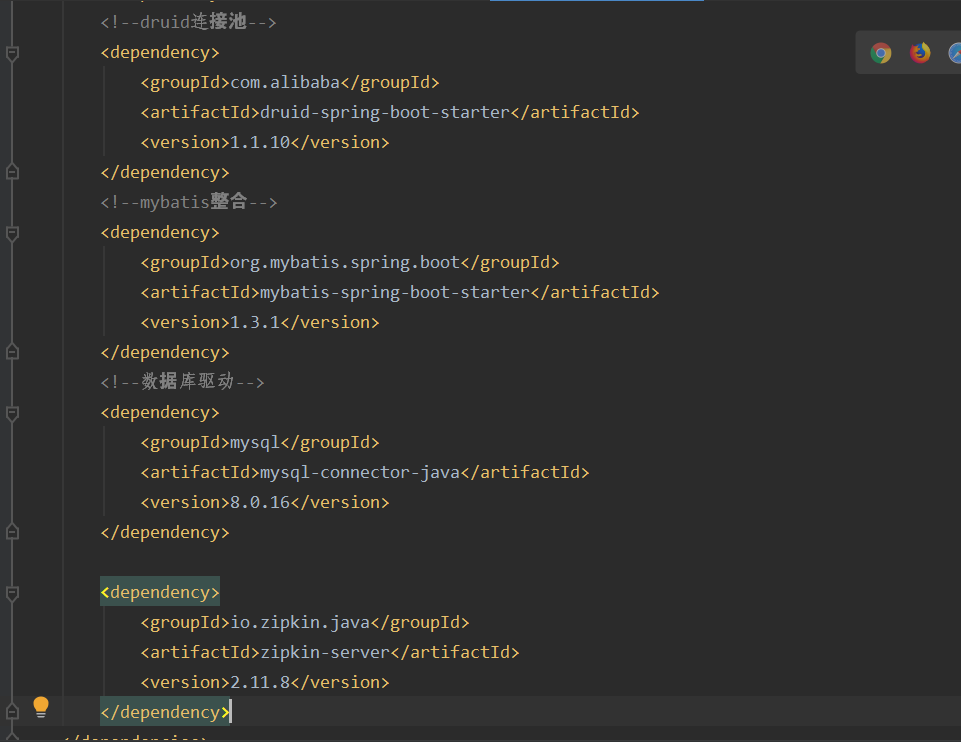


## **4.2.3 提供者模块**

提供者模块主要包括menu、user、order，client通过调用这些提供者的接口进行实现功能

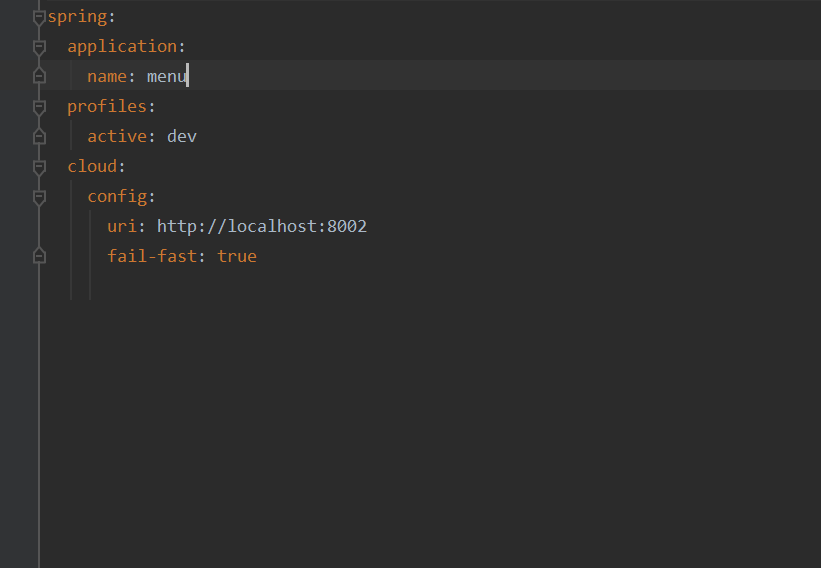
1. 项目搭建好后menu,order,user的pom.xml文件代码如下所示：

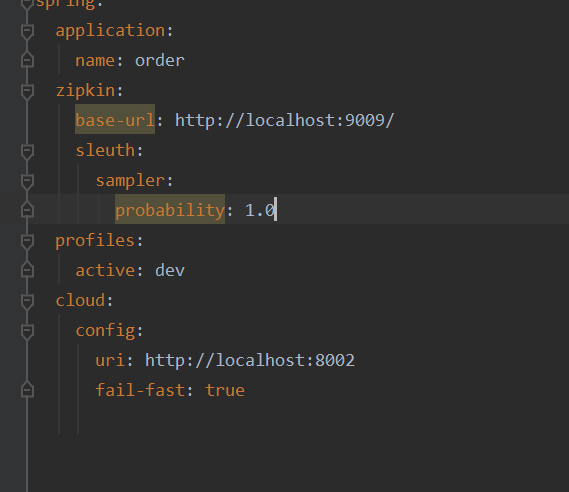


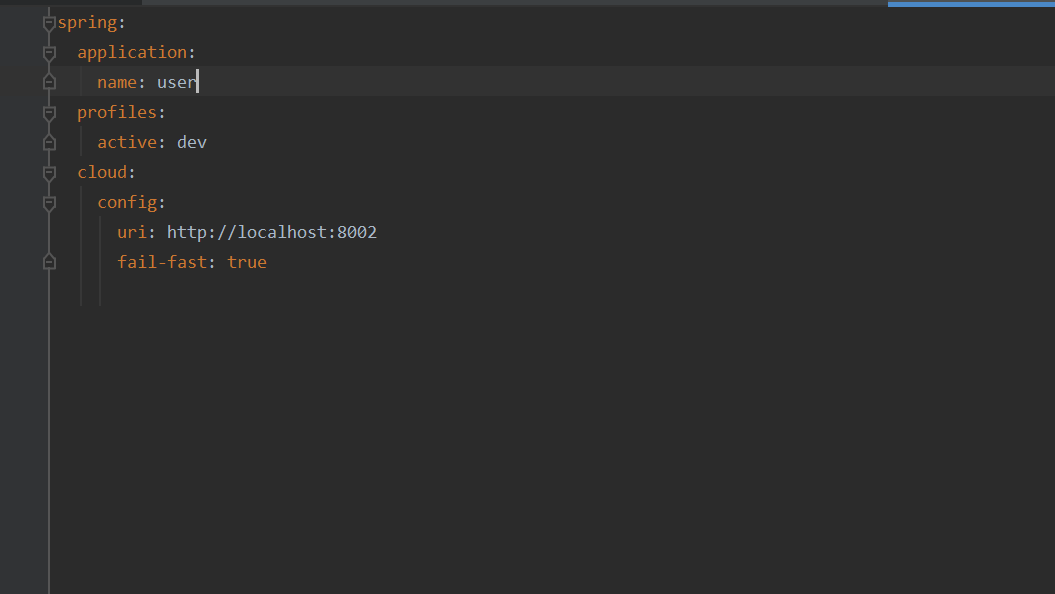




(2)配置文件如下：







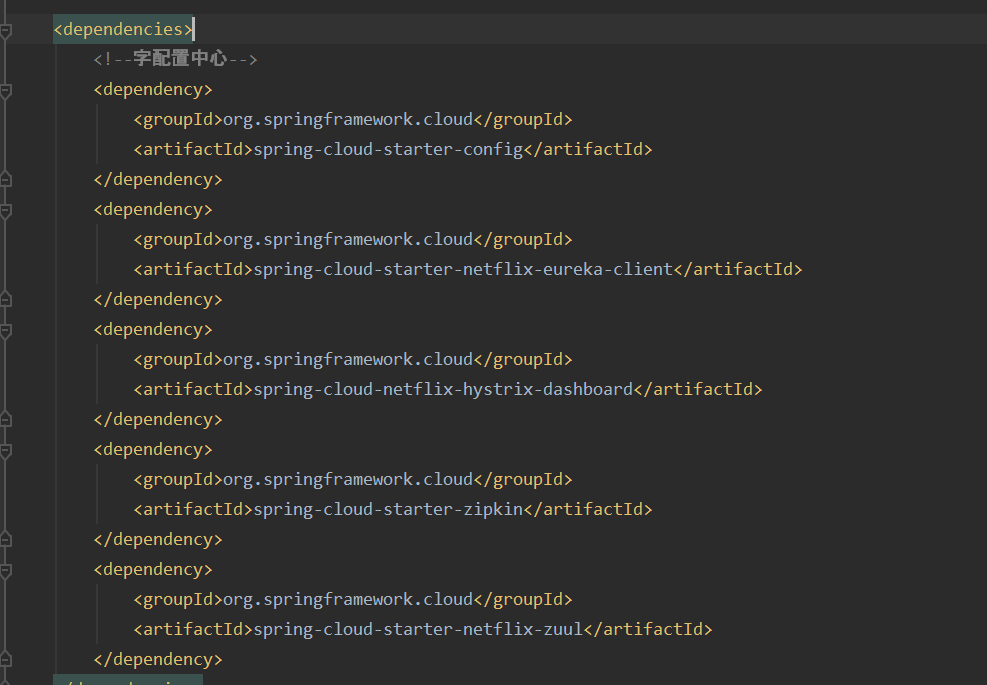
实现效果;



**4.2.4 网关模块gateway-zuul**

gateway-zuul是网关模块，它会过滤所有来自外部链接的请求，并对请求头中附带的信息进行判别，然后验证信息，并作相应的处理。

1. 项目搭建好后的pom.xml文件代码如下所示：

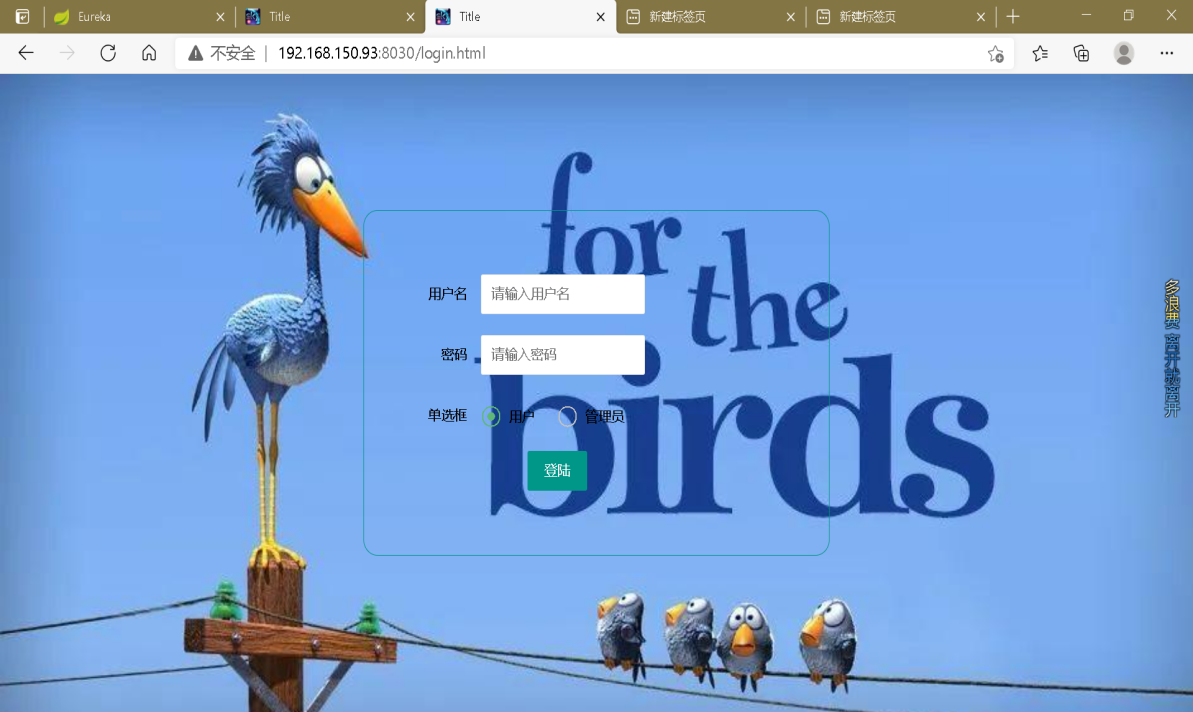


1. 配置文件如下：



(3)启动gateway-zuul模块，调用其中 allclient接口，实现的效果如下：





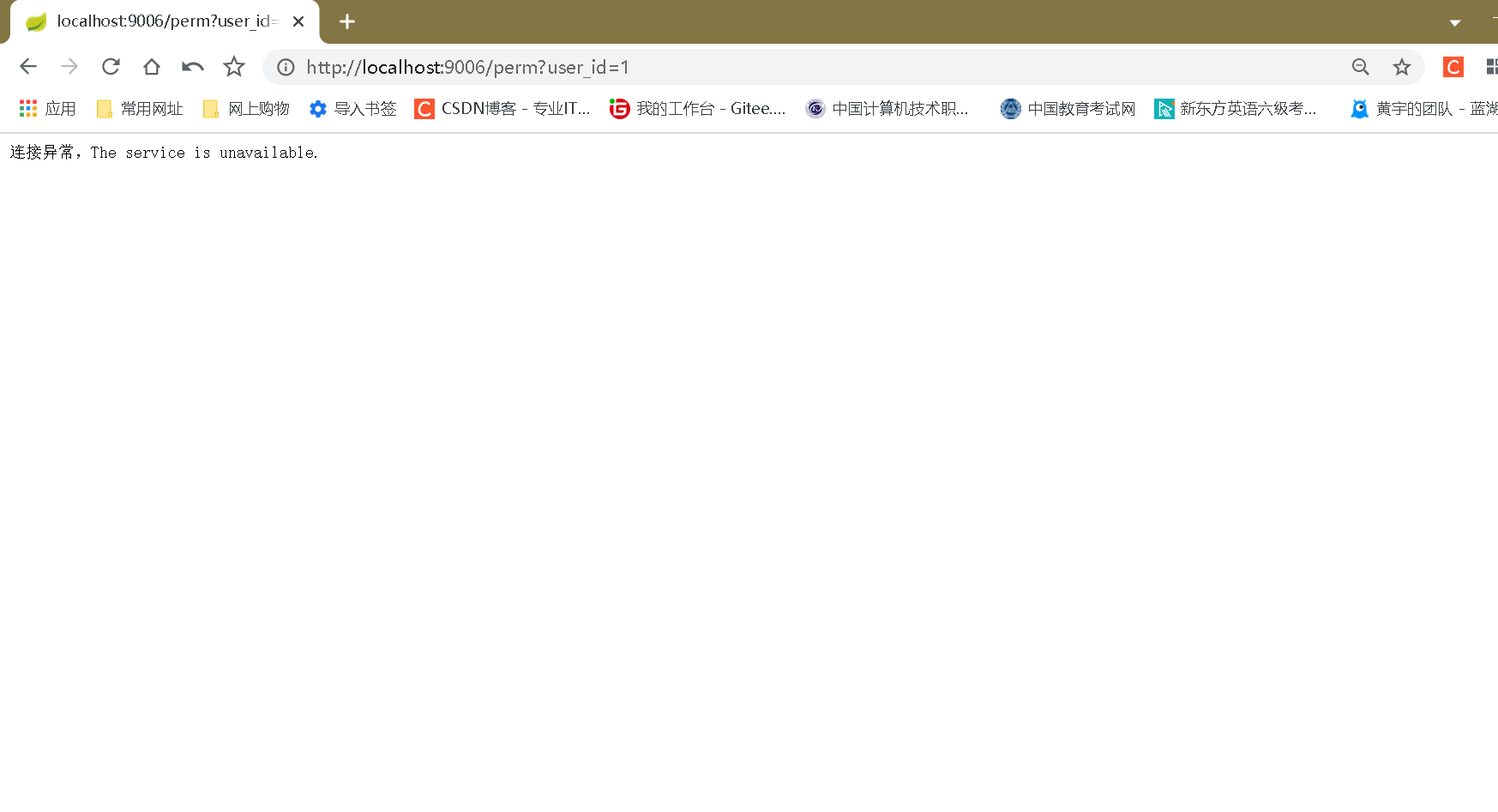
自定义网关zuul过滤器;



**Zuul与Hystrix结合实现熔断，实现熔断功能时需要完Fallbackprovider**

**接口，代码截图如下：**

启动eureka-sever和gateway-zuul项目，访问浏览器http://localhost:9006/user\_id,显示的结果如下：



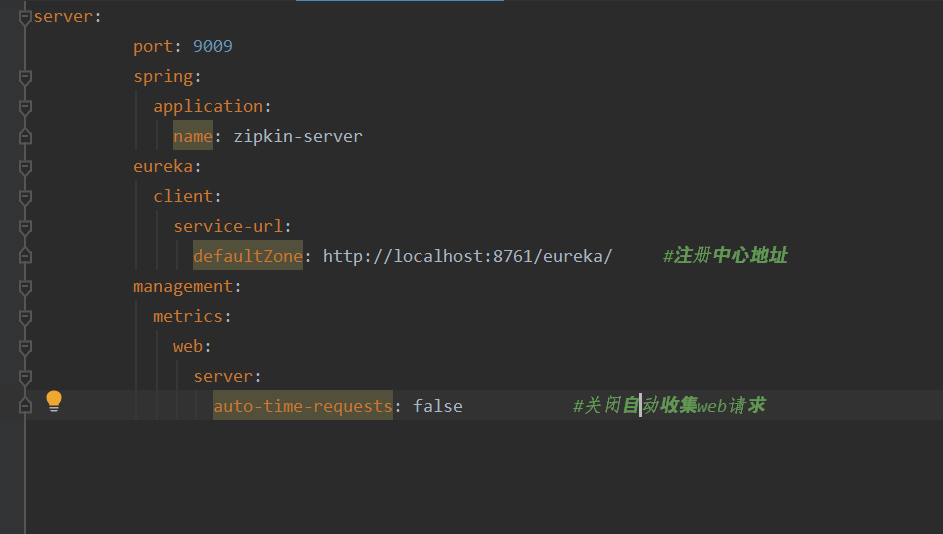
## **4.2.5 请求链路追踪模块zipkin-server**

zipkin-server是请求链路追踪模块，它会追踪所有的通过网关的请求，并把追踪到的链路信息显示到网页上。在本项目中，zipkin-server模块会追踪透过网关的从gateway-zuul到allcient的请求链路。

1. zipkin-serve项目搭建好后的pom.xml文件代码如下所示：



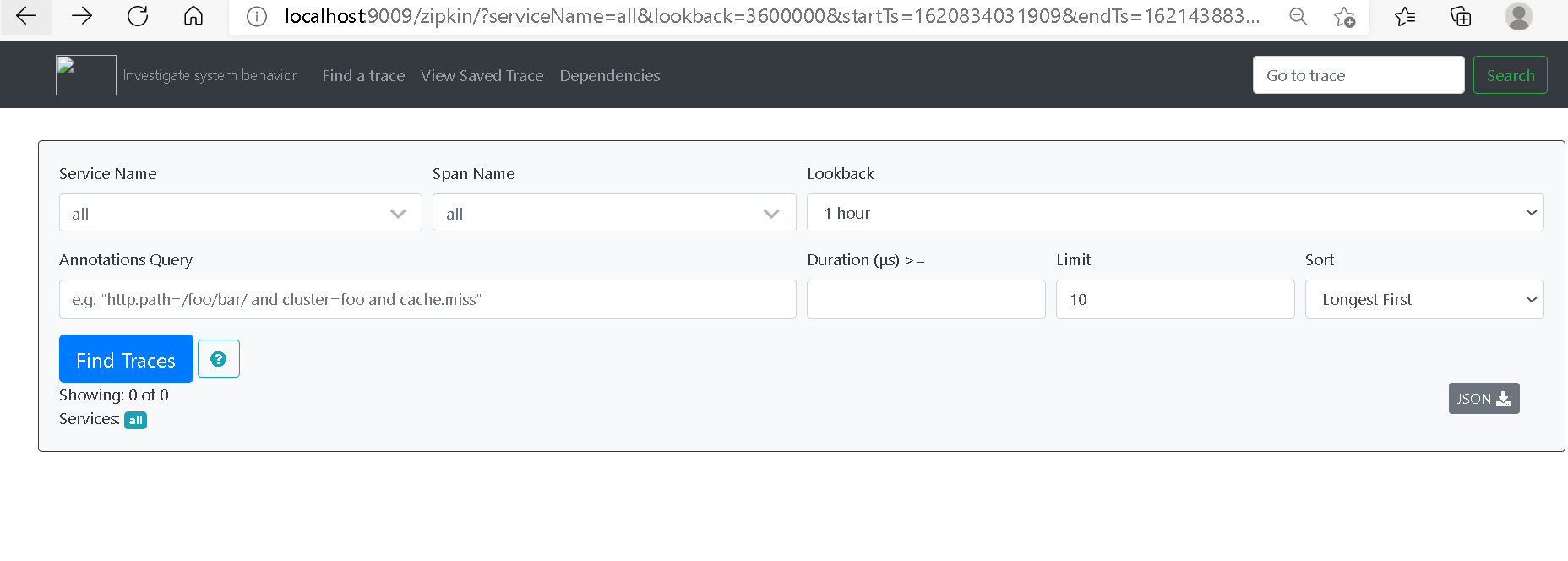
1. 配置文件如下：



（3）同时运行allcient以及开启网关服务gateway-zuul，并开启zipkin-sever项目，使用浏览器访问：

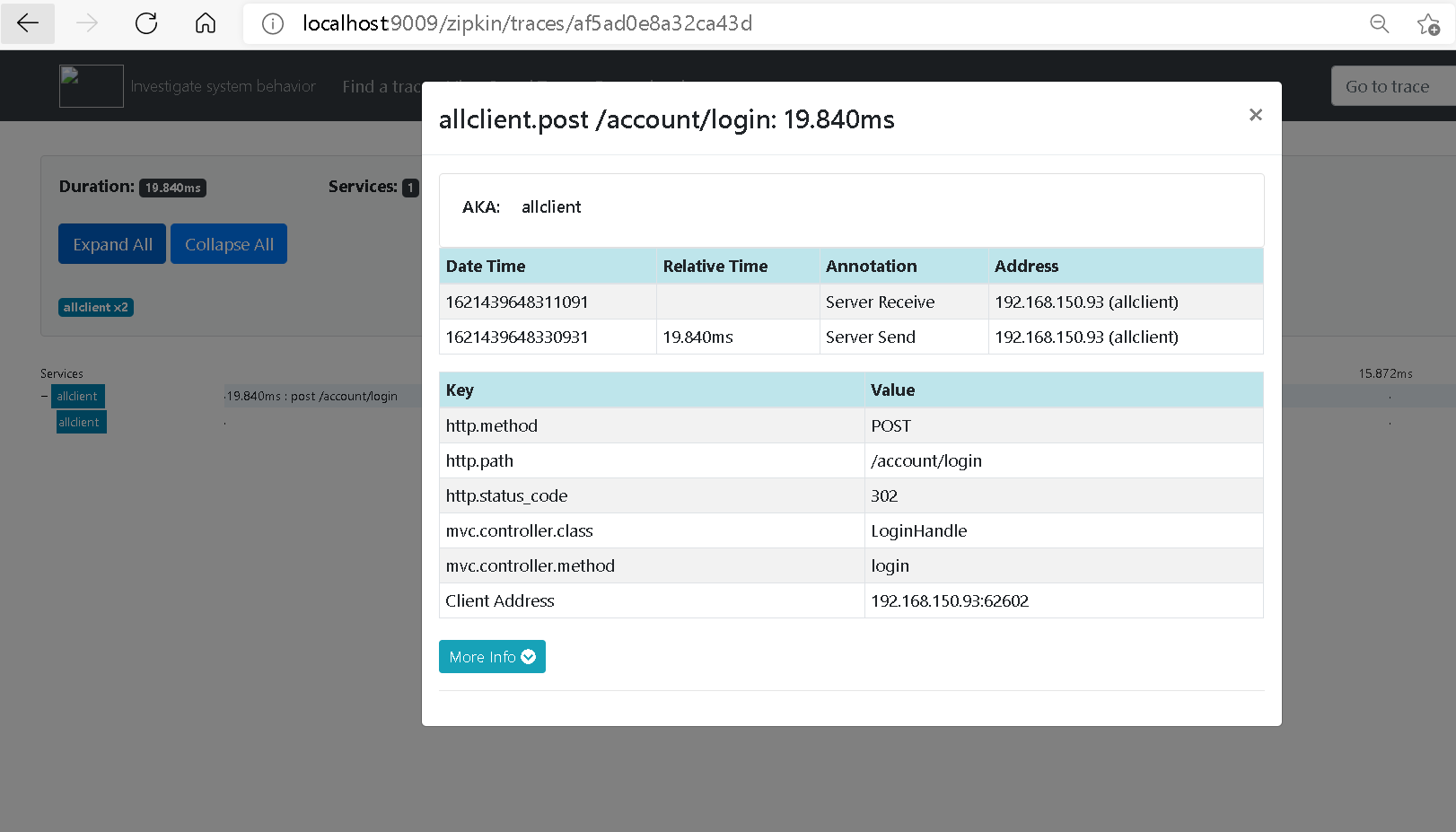
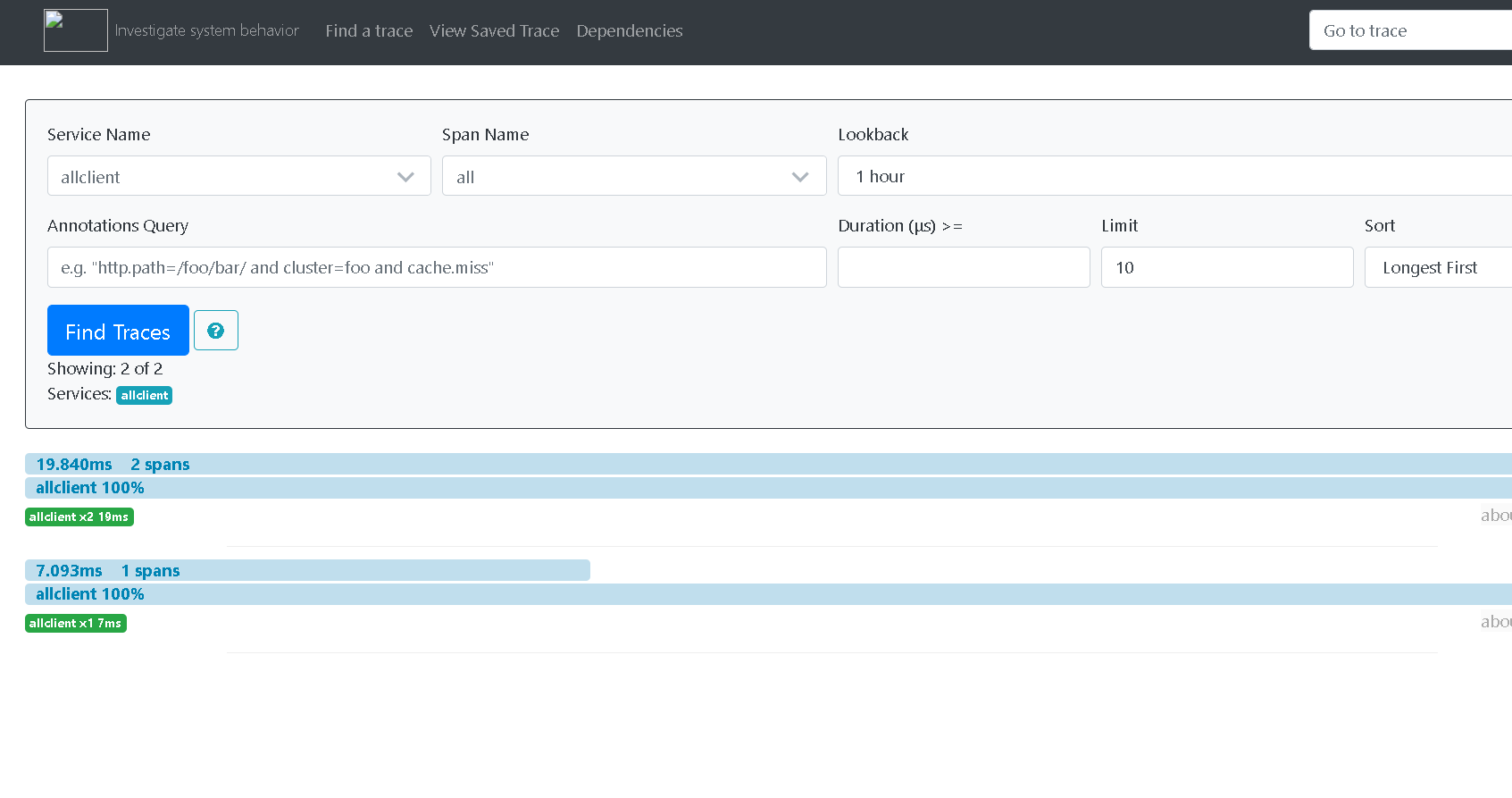
<http://localhost:9009/zipkin/>

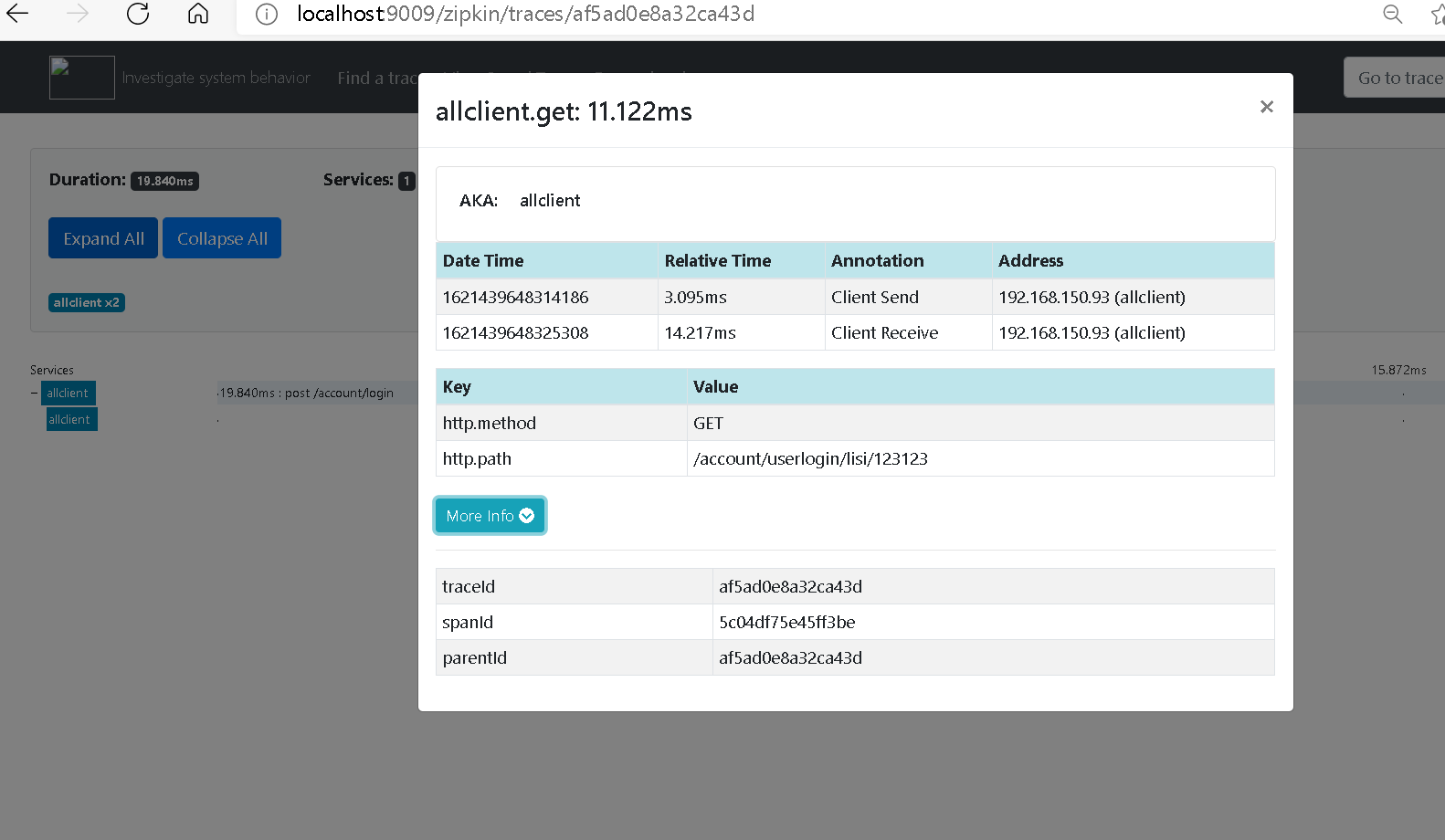
页面效果如下所示：



截图可以看到三个项目都已经被追踪，点击按钮“Find Traces”，查看gateway-zuul,

allclient之间的调用记录，如下所示：





## **4.3 系统的开发平台及运行环境**

## **4.3.1系统的开发平台**

系统的开发是在Tomcat环境下进行的，利用IntelliIDEA+2019.3+Apache+mysql的工具，以及Bootstrap+JQuery ui框架来实现页面。Tomcat服务器是一个开源的轻量级web应用服务器，广泛运用于中小型系统和并发量少的场合下。它被认为是Servlet和JSP程序开发和调试的首选。

Tomcat的环境主要有以下几方面技术优势：

1．Tomcat中使用的是WAR文件，相当于一个压缩包。

2．将WAR存放到Tomcat的webapp目录下jsp文件可自动检测并解压。

3．Tomcat既有处理HTML的功能，也可以作为ssm的容器。

## **4.3.2系统的运行环境**

操作系统：Windows 10以上版本。

服务器软件tomcat+mysql.8.0以上版本。

浏览器：IE、Fire Fox、Google Chrome、360浏览器。

# **第5章** 系统性能

## **5.1 性能需求**

（1）系统最小寿命 ：

系统应该能在 无重大改动的条件下正常运行。

（2）对设备费的要求:

要求计算机稳定性良好，整套系统经济实惠。。

1. 可以利用的信息和资源：

部门已经配置所有的计算机，以及网络设备，操作人员将使用智能操作。

1. 用户以及管理员有计算机方面的培训。

## **5.2 安全性需求**

由于微服务书籍管理系统的书籍信息量大，所有在对这些书籍信息导入 和查询要保证速度，在用户管理过程中又要保证事务的完整性，对于整个系统 ，需要完整的权限控制，防止被恶意攻击，修改原始 记录，另外，系统要求用户在登陆时需要身份验证。

## **5.3软件质量属性**

1. 可使用性 :简捷方便，性能优越，用户使用方便 。
2. 保密性 :用户需输入账号密码登录验证，便可查询本人的基本信息情况。

可维护性 ：随时监控，发现问题及时解决，维护性强。

**第6章** 系统测试

## **6.1 系统测试**

### 6.1.1测试的意义

测试在系统开发过程中是最轻易被忽视的一个步骤，但其实测试在系统开发中是不可或缺的，对系统的开发质量起着关键的作用。

在系统开发过程中，我们无法避免会有大大小小的bug，语法上的bug主要在代码中体现出来，而一些逻辑错误以及小错误和问题，都需要从测试这个过程来发现错误及问题的所在之处，以便对系统的修改和完善，提高系统的稳定性和健壮性。

在软件的生命周期中，测试是其中极其关键的一道工序，对开发高质量的系统有着重要的意义。

### 6.1.2测试的目的

在系统简单运行的情况下，着手对系统进行简单地检测，尽早发现系统的问题和漏洞，对bug进行修改，对系统进行完善，为做好一个成功的系统打下基础。

软件测试的根本目的是满足客户需求，它的过程就是给软件找出各种BUG，但我们为什么要找出这些漏洞呢？就是为了让客户体验更好。软件测试定义发生了改变，测试不单纯是一个发现错误的过程，而且将测试作为软件质量保证(SQA)的主要职能。其目的在于检验它是否满足规定的需求或弄清预期结果与实际结果之间的差别。

通过测试的系统，不说100%没有漏洞，但是测试可以提高其系统的稳定性和健壮性。

6.1.3软件测试方法

根据测试对象和内部结构可分为：黑盒测试和白盒测试。

* 根据是否运行程序可分为：静态测试和动态测试。
* 根据系统开发历程可分为：单元测试、集成测试、确认测试、系统测试、验收测试、回归测试

动态测试方法：指通过运行软件来检验软件的动态行为行和运行结果的正确性。具体操作就是输入相应的测试数据，检查输出结果和预期结果是否相符的过程。根据动态测试在软件开发过程中所处的阶段和作用，动态测试可分为如下几个步骤：  
（1）单元测试  
 单元测试是对软件中的基本组成单位进行测试，其目的是检验软件基本组成单位的正确性。在公司的质量控制体系中，单元测试由产品组在软件提交测试部前完成。单元测试是白盒测试。  
 白盒测试又称结构测试、透明盒测试、逻辑驱动测试或基于代码的测试。白盒测试是一种测试用例设计方法，盒子指的是被测试的软件，白盒指的是盒子是可视的，你清楚盒子内部的东西以及里面是如何运作的。"白盒"法全面了解程序内部逻辑结构、对所有逻辑路径进行测试。"白盒"法是穷举路径测试。在使用这一方案时，测试者必须检查程序的内部结构，从检查程序的逻辑着手，得出测试数据。    
（2）集成测试  
 集成测试是在软件系统集成过程中所进行的测试，其主要目的是检查软件单位之间的接口是否正确。在实际工作中，我们把集成测试分为若干次的组装测试和确认测试。  
组装测试，是单元测试的延伸，除对软件基本组成单位的测试外，还需增加对相互联系模块之间接口的测试。如三维算量软件中，构件布置和构件工程量计算是软件不同的组成单位，但构件工程量计算的数据直接来源于构件布置，两者单独进行单元测试，可能都很正常，但构件布置的数据是否能够正常传递给工程量计算，则必须通过组装测试的检验。集成测试是白盒测试。  
确认测试，是对组装测试结果的检验，主要目的是尽可能的排除单元测试、组装测试中发现的错误。  
（3）系统测试  
 系统测试是对已经集成好的软件系统进行彻底的测试，以验证软件系统的正确性和性能等满足其规约所指定的要求。系统测试应该按照测试计划进行，其输入、输出和其它动态运行行为应该与软件规约进行对比，同时测试软件的强壮性和易用性。如果软件规约（即软件的设计说明书、软件需求说明书等文档）不完备，系统测试更多的是依赖测试人员的工作经验和判断，这样的测试是不充分的。系统测试是黑盒测试。  
（4）验收测试  
 这是软件在投入使用之前的最后测试。是购买者对软件的试用过程。在公司实际工作中，通常是采用请客户试用或发布Beta版软件来实现。验收测试是黑盒测试。  
黑盒测试也称功能测试，它是通过测试来检测每个功能是否都能正常使用。在测试中，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，在程序接口进行测试，它只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数据而产生正确的输出信息。黑盒测试着眼于程序外部结构，不考虑内部逻辑结构，主要针对软件界面和软件功能进行测试。     
黑盒测试是以用户的角度，从输入数据与输出数据的对应关系出发进行测试的。很明显，如果外部特性本身设计有问题或规格说明的规定有误，用黑盒测试方法是发现不了的。     
（5）回归测试  
 即软件维护阶段，其目的是对验收测试结果进行验证和修改。在实际应用中，对客诉的处理就是回归测试的一种体现。