|  |
| --- |
| **基于分布式系统开发的校园闲置交易平台设计** |

|  |
| --- |
| Design of Campus Idle Trading Platform  Based on Distributed System Development |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 学生姓名 | 罗鸿伟 | 学号 | | 201611921120 | | |
| 所在学院 | 数学与计算机学院 | | 班级 | | | 计数1165 |
| 所在专业 | 信息与计算科学 | | | | | |
| 申请学位 | 理学学士 | | | | | |
| 指导教师 | 陈入云 | | 职称 | | 教授 | |
| 副指导教师 |  | | 职称 | |  | |
| 答辩时间 | 年 月 日 | | | | | |

**目 录**

[设计总说明 I](#_Toc41599378)

[InTRODUCTION II](#_Toc41599379)

[1 绪论 3](#_Toc41599380)

[1.1 设计背景及需求 3](#_Toc41599381)

[1.2 架构选型原因及其优缺点 3](#_Toc41599382)

[1.2.1 架构选型原因 3](#_Toc41599386)

[1.2.2 优缺点 4](#_Toc41599387)

[2 开发环境工具和技术要点 4](#_Toc41599388)

[2.1 开发环境工具 4](#_Toc41599389)

[2.2 技术要点 4](#_Toc41599390)

[3 可行性分析和功能确定 6](#_Toc41599391)

[3.1 可行性分析 6](#_Toc41599392)

[3.2 主要功能确定 6](#_Toc41599393)

[4 平台开发的基础搭建 7](#_Toc41599394)

[4.1 数据库建立 7](#_Toc41599395)

[4.2 前端UI设计 8](#_Toc41599396)

[4.3 项目结构搭建 8](#_Toc41599397)

[4.4 依赖导入 9](#_Toc41599398)

[4.5 数据库连接 10](#_Toc41599399)

[4.6 目录结构 11](#_Toc41599400)

[5 部分功能介绍以及代码实现 12](#_Toc41599401)

[5.1登录页面 12](#_Toc41599402)

[5.2用户注册界面 15](#_Toc41599403)

[5.3首页展示 16](#_Toc41599404)

[5.4个人信息修改功能 17](#_Toc41599405)

[5.5上传个人闲置物品信息 17](#_Toc41599406)

[5.6 功能总结 19](#_Toc41599407)

[6 分布式服务注册和和管理测试 19](#_Toc41599408)

[6.1 注册 19](#_Toc41599409)

[6.2 流量守卫兵 20](#_Toc41599410)

[7 项目总结 22](#_Toc41599411)

[鸣 谢 23](#_Toc41599412)

[参考文献 24](#_Toc41599413)

# 设计总说明

当今的互联网已充斥到我们生活的每个角落，互联网的利弊，既为我们的生活带来了很多便利，也带来了新的困扰。在大学的四年间，通过自身的生活经历以及观察周围的同学的状况，发现许许多多同学包括本人随着时间的累计，总会日积月累下很多闲置的用品，平时少用，却弃之可惜，又霸占着许多空间，亦有许多同学想要购买一些使用寿命较长物品，却因为考虑到在校时间不长，买了也许无法合理利用其价值，导致金钱和资源的浪费而止步；得于这种广泛的现象，和受到互联网的影响，我便想到利用好这把互联网的双刃剑，通过搭建一个互联网校园闲置交易平台，仅提供给本校的学生使用，可以让校园师生通过该平台出售自己的闲置闲置物品，也可以通过该平台购买一些别人闲置的，物美价廉又适合自己的闲置物品。这种二手平台如今市场上已经有了很多，但是由于是面向社会上所有消费人群的，信用度不尽人意，这是为什么身边的同学极少有人使用这些平台的原因，也是为什么本人设计该产品仅为本校学生使用的原因，也能从侧面反映出该产品的可行性和实用性。

本校园闲置物品平台为了保证消费者的利益安全，以及防止非本校师生利用该平台进行一些非法的交易买卖，初步的设计理念为，该平台不为消费者代收任何费用，包括买卖交易之间的金额代收，实现价格全透明、实物交易、买卖双方之间不存在中间商；而平台则主要为本校师生们提供上传闲置物品信息，包括但不限于闲置物品照片、原购买价格、现售出价格、成色、卖家联系方式等；师生们既可是卖家，亦可是买家，通过该平台浏览闲置物品，物色合适自己的商品，进而联系卖家，进行交易。

本毕业设计以产品设计和代码实现为主，不仅锻炼自身对产品需求的分析和设计能力，更能学习使用的代码编写，掌握编程技术。

关键词：微服务、分布式、Java、H5、校园闲

# InTRODUCTION

Nowadays, the influence of the internet is full in our life. The advantages and disadvantages of the internet have not only brought us convenience, but also brought the new troubles. During four years at university, through my own experience and observing the students around me, I found that many students including myself had more and more unused goods as the time goes on which hardly used but reluctant to discard, and take in a lot of space. Some students want to buy something which can use for a long time, but because of considering that the time in school is very short, these somethings might not be able to make good use of its value. Under the influence of the internet, I wanted to make good use of the convenience from the internet, design a network platform to deal with the problem of idle items, which only for the students and teachers who in the same school .The students not only can make campus through the platform to sell their idle items, but also can through this platform to buy something which they want and cheaper. Today there have been many second-hand platform on the market, however they exist the credit problems because they are available to all people in the society. That is why few students use them, and it is the reason why I design the product which only for students to use too. On the other hand , it reflects the feasibility and usability of the platform .

In order to protect the interests of consumers, and to prevent someone to use the platform for some illegal transactions, the design concept for the platform is not collection any charge for the consumer, including the amount of transactions between the collecting, realize transparent price, barter trade, no intermediaries between buyers and sellers; The platform mainly provides teachers and students of the same university to upload the information for their unused goods, including but not limited to photos of unused goods, original purchase price, current sale price, and the seller's contact information. Teachers and students are both sellers and buyers. Through the platform, they sell idle items, find suitable commodities, and then finish the transaction by themselves.

This graduation project is based on product design and code implementation, it not only exercises my ability to analyze and design product requirements, but also enables me to learn program and master programming techniques.

Keywords: microservice, distributed, Java, H5, campus idle

**基于分布式系统开发的校园二手交易平台设计**

信息与计算科学，201611921120，罗鸿伟

指导教师：陈入云

# 1 绪论

## 1.1 设计背景及需求

当今时代，网上购物以及成为了潮流，尤其是在当代的大学生人群中最为流行。在双十一、双十二、618等大型的网购节日以及平时的各种新颖丰富的推销手段，使不少大学生产生了许许多多不合理消费，许多消费产品仅仅只是一时兴起，或者贪图减价而购买但购买后却几乎从未使用，亦或者许多物品使用后觉得不合适，但难以退货；更或者只为了临时使用，不得不购买，但使用后只能闲置，弃之可惜的物品。在大学四年期间，随着时间的累计，闲置物品越积越多，这不仅增加了同学们的经济压力，更是导致物品资源得不到合理的利用导致的资源浪费。

设计校园二手交易平台的意义在于，它不同于市场上现有的二手交易平台（app，网站等），因为该平台仅为同校区的师生提供服务，因此，在双方交易时，可以直接选择面对面进行验货交易，以及根据货物的成色进行具体的价格商议，从根源上避免了中间商赚差价、货物造假等现象。这更加符合校园师生的使用需求，吸引用户使用，不仅能为同学们处理闲置物品，同学也可以在该平台物色合适自己，价格便宜的商品，作为学生，这能很大程度上缓解我们的经济压力，更重要的是能使物品能充分发挥自身的利用价值。

## 1.2 架构选型原因及其优缺点



### 架构选型原因

本校园闲置平台使用了目前IT市场上面使用的相当流行的开发方式--分布式系统开发，这种开发方式主要是基于一个开发架构--微服务架构[1]。微服务[2] 架构是一种架构思想，架构就是为了解耦，主要的开发方式是分布式系统开发。所谓的分布式系统开发就是把以往的单体应用、单体系统按照功能模块拆分出多个子模块，子系统，目的是为了可以提高系统的可用性，应对高并发场景。通过分布式系统开发方式开发的项目由于把一整个系统拆分为多个子系统，这样可以保证当某个子系统出了问题而不会导致整个系统不可用的场景，从而大大提高系统的可用性；再针对某个子系统进行集群搭建，进而在高并发，多人同时使用的情况下，对请求进行分流给不同的服务器，缓解服务器的压力，尽可能避免服务器崩溃。微服务架构是当今中国最大电商阿里巴巴公司的淘宝软件采用的软件架构，并在阿里巴巴技术人员的开发使用下诞生了一套实现微服务架构的开发生态—spring cloud Alibaba[3]。淘宝软件在应对百亿级别并发量的双十一大型高并发场景足以证明微服务架构的强大之处，在考虑到互联网的不断发展，使用互联网的人越来越多，网络并发量的暴增是势不可挡的，所以决定采用微服务架构作为校园闲置平台的开发架构思想，即分布式系统开发方式。

### 优缺点

分布式系统开发的优点[4]：使开发的程序拥有更高的性能、应对更高的并发量、以及程序解耦，对程序的拆分使开发更专注高效、模块的独立运行能够更合理地分配硬件资源。

分布式系统开发的缺点：服务多了，服务之间的通信如何解决、服务如何治理、用户如何访问具体服务、某个服务挂了怎么办。这四大问题从根本上反应了微服务的问题，而且拆分出来的服务越多，对整个服务的部署也是很大的问题，关键是目前仍旧没有快捷的方法解决复杂的微服务的四大难题。

# 2 开发环境工具和技术要点

## 2.1 开发环境工具

校园闲置平台开发环境使用JDK1.8[5]，代码编译软件使用Intellij IDEA，数据库为MySQL，

数据库连接工具使用了开源版的 Navicat Premium 12，使用了finalshell作为服务器连接工具，测试软件为postman。

## 2.2 技术要点

Spring + Spring MVC + MyBatis（SSM）

Spring是一个轻量的、企业级的JavaEE开发框架[6]，如今，Java语言能广泛流行，Spring可谓是最大功臣。Spring框架亦称为胶水框架，他的强大之处就是可以作为一个基础模板，通过粘合上其他各式各样的框架融汇在一起，方便开发者使用；Spring也是个容器，他代替了开发者管理许许多多的Java对象（Java是门面向对象的开发语言）。Spring的核心点为IOC和AOP，IOC为控制反转，也是Spring容器能够为开发者管理Java对象的核心；AOP为面向切面编程思想，通过使用AOP思想可以方便我们实现对事务的管理。

SpringMVC是在使用Spring的基础上的一个三层架构，它把一整个程序分为：模型（Model）、视图（View）和控制器（Controller）。SpringMVC为开发者开发Web应用程序提供全面的基础架构支持，使得Java开发更加方便和快捷。

MyBatis为数据库持久层框架，是Java操作数据库的框架之一，在国内较为流行，与他有着相同作用的是Hibernate框架。Hibernate在国外普遍使用（作为Java的数据库持久层框架），其具有高效的自动化和高性能的二级缓存；相当于Hibernate，MyBatis则具有更好的灵活性，更加适合应对多变的需求环境。

Spring + SpringMVC + MyBatis的结合使用既是广为流行的SSM开发框架，是Java用于Web开发的核心，也是目前国内最为流行的Java开发框架之一，也是本校园闲置平台所采用的基础框架。

1. Spring Boot + Spring Cloud Alibaba [7]

Spring Boot是新一代JavaEE开发标准，使用SpringBoot能为我们省去许许多多以往繁琐的配置文件（约定优于配置），能实现程序的秒级启动，简化开发者的开发步骤。

Spring cloud是一个生态，也可以说是一个接口，他就是为了解决上述的微服务带了的四大难题，他提供了一整套的接口和方法给开发者使用。目前最为流行使用的是netfix（网飞公司）和阿里巴巴整理出的一整套的具体使用标准，分别为Spring Cloud Nefix 和Spring Cloud alibaba。（由于主要使用的是阿里巴巴的Spring Cloud Alibaba，便介绍Spring Cloud Alibaba 以下简称SCA） SCA提供了服务注册中心Nacos、Dubbo（先已经给阿帕奇公司维护）、sentinel（流量守卫兵），网关gateway，分别针对解决服务如何治理、服务如何通信、服务挂了怎么办、用户如何访问具体的服务的问题。

Spring Security

Spring Security安全框架在本项目中主要是用来做用户的登录管理，管理用户的登录验证以及控制用户的多设备登录以及打算用于短信验证码登录。

1. Html、css、JavaScript、bootStrap、jQuery、MySQL[10]

以上均为用于前端开发，其中html、css、JavaScript、jQuery是前端开发的主要开发技术[9]，bootstrap为组件库，为我们提供了许许多多现成的组件，我们可以直接在官网上获取对应的样式代码，实现前端的快速开发。

Mysql[10]作为数据库语言，实现对数据库的操作。

# 3 可行性分析和功能确定

## 3.1 可行性分析

1. 用户量可行性分析

本闲置平台初衷为海大校区的校友们提供一个出售和购买闲置物品的物联网网站，从本人在校期间，与同学的交流和走访得知，同学们基本上都会有不少闲置的物品，并且随着在校时间的增加，闲置的物品也越多，尤其到了毕业季，几乎每个应届毕业生都为处理这些闲置物品而感到烦恼，要么通过变为废品低价出售，花费更多的物流费用寄回家中，甚至不少学生直接丢弃，导致资源的浪费。因此从用户需求量来说，是完全可行的。再到今后，我们可以对平台进行升级，推广，使该平台可以发展推广到其他校园，成为不同校园的专属闲置平台，用户量便能随之增加。

1. 技术可行性分析

本平台采用的开发技术均为现在市场上成熟而广泛的开发技术，为了开发该平台，我学习了Java的基础语法[11]、spring、spring mvc、MyBatis等技术框架以及微服务的spring cloud albaba的相关组件和前端的html、css、js、jq，对平台所用的技术都有了一定程度的掌握，加上所用的技术均为市面流行的技术，技术人员众多，开发难度不大。

1. 开发成本可行性分析

由于本项目由本人独立开发，唯一需要购买的是产品上线所需要的服务器（平均400人民币一年，具体看相关配置和报价），其中开发过程中使用到的开发软件和工具都是不收钱的，所需要的开发费用很少，本人完全可以独立支持。

## 3.2 主要功能确定

1. 用户信息验证功能

为了规范用户合理合法使用，记录用户交易信息，登录注册功能是必不可少的，基础版本提供用户名和密码注册模式，并在日后逐渐开发提供手机验证注册和微信号注册；为已经注册的用户提供登录页面，通过对用户输入的信息进行校验，完成用户的登录，并提供免登陆功能，当用户选择免登陆时，可以实现指定时间范围内无需登陆即可。

闲置分类功能

闲置分类功能是用于对各种各样的闲置物品进行分类，例如体育器材、数码电子、娱乐玩具、衣服装饰等，方便购买者寻找自己所需的物品。

1. 分页功能

分页功能是在当数据量达到一定程度时，使数据进行分页展示，使页面更加美观，闲置物品的展示更加的规范化，提高用户的使用体验。

1. 联系卖家功能

该功能为消费者找到合适自己的闲置物品时，联系卖家，买卖双方通过交流获取双方的微信号等详细交流方式，进而完成最终的交易。

1. 上传闲置物品功能

用户可以通过上传闲置物品界面窗口，上传闲置物品的部分相关信息，包括照片、名字、简介、价格等，用户通过上传自己的闲置物品，最终展示到闲置物品列表，提供消费者选择购买。

1. 个人信息管理功能

用户可以对自己的用户名，密码以及头像和个人信息进行修改。

1. 个人物品管理功能

该功能使用户可以对自己上传的闲置物品信息进行修改，以及下架已经上传的个人闲置物品。

# 4 平台开发的基础搭建

## 4.1 数据库建立

使用MySQL5.6建立secondHangSchool数据库，包含用于储存闲置物品的good表，用于储存用户信息的user表， manage表主要是用来用于储存管理员一些个人信息和账号密码的， diss表主要是用于记录用户对物品作出的评论的， type表主要是用于记录闲置物品类型的，其包含的字段和字段类型如下：图1数据库设计

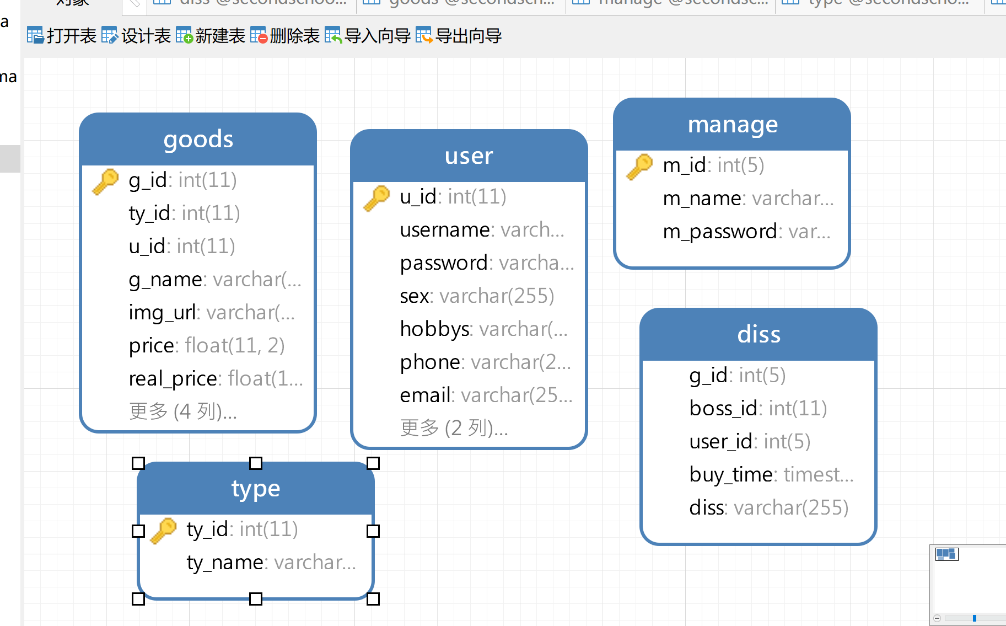
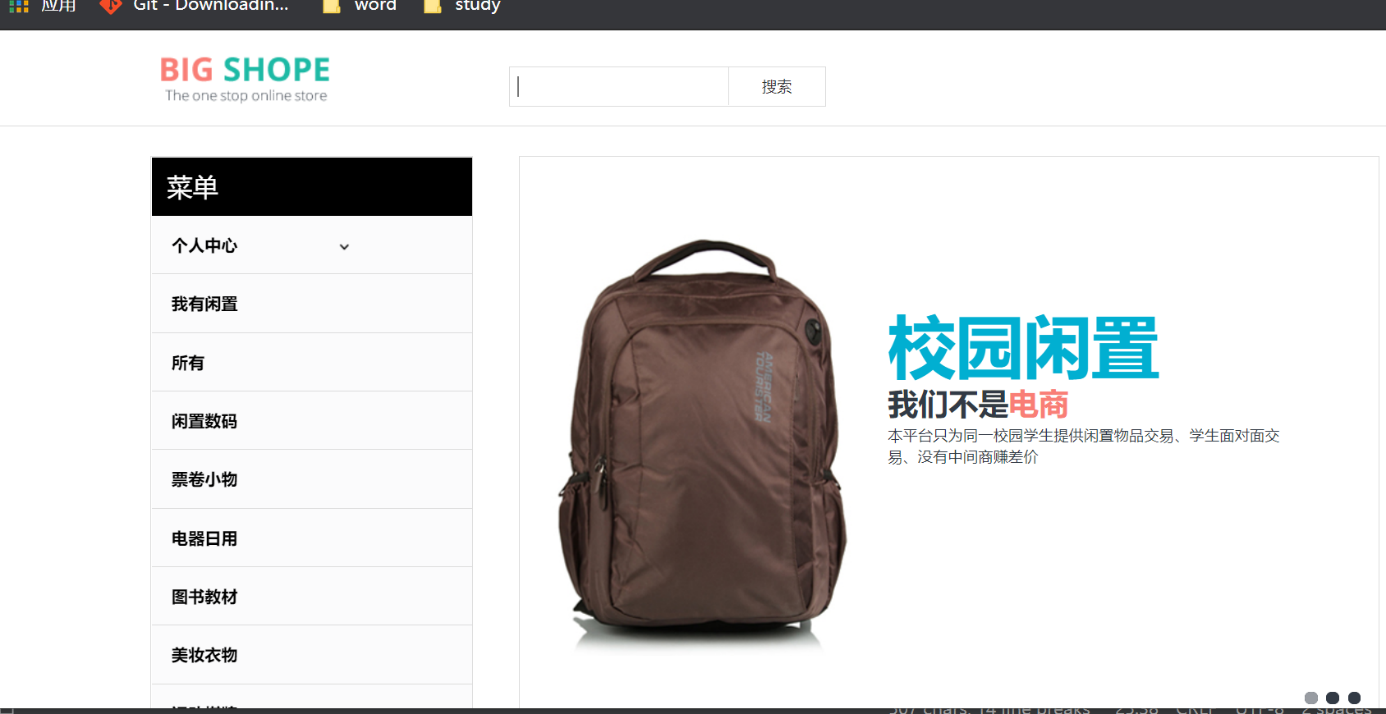


图1 数据库设计

图2 总体流程设计图

## 4.2 前端UI设计

前端UI界面的设计包括设计登录页面的样式、注册页面设计、首页页面设计，设计理念为简约设计，整体页面干净简约，用户操作难度低，用户与页面交互友好。实现的技术语言均为目前市面主流的前端技术（html5、css、js），学习并不困难，而且方便维护，通用性高。



## 4.3 项目结构搭建

按照分布式系统开发的设计理念[12]，在项目结构上，主要分为统一依赖管理模块、服务消费者模块、服务提供者模块、网关模块。

统一依赖管理模块主要的作用是管理其他模块所需要用到的第三方依赖包、工具包、jar包的版本号管理，这里主要是利用了Java面向对象编程思想的重要特性之一——封装，把所有需要用到的依赖的版本号抽取出来，统一管理，如果需要修改，则只需要修改统一依赖管理模块里面的pom.xml文件里面对应的依赖包的版本号，则可实现对所有用到这依赖包的版本号进行修改。

服务消费者主要是接受前端的请求，为前端提供接口的调用，与前端进行交互，但他不参与接口的具体业务实现，主要是消费服务提供者模块的接口实现业务返回的结果，并且对返回的结果进行判断和处理，最后返回给前端；服务提供者模块则主要是编写关于业务逻辑层的代码编写，实现与数据库的连接和对数据库的操作，不参与与前端的交互。这样设计的目的是为了解耦，同时通过模块的拆分，使多个服务消费者可以使用同一个服务提供者处理出来的结果，这就大大提高了代码的可用性，减少重复代码的书写，更重要的是，服务消费者调用不同的服务提供者，即使某一个服务提供者模块出了问题，也不会影响到其他模块的运行，这也就是分布式系统分开的具体表现之一。

主要功能模块如：图3项目主干结构

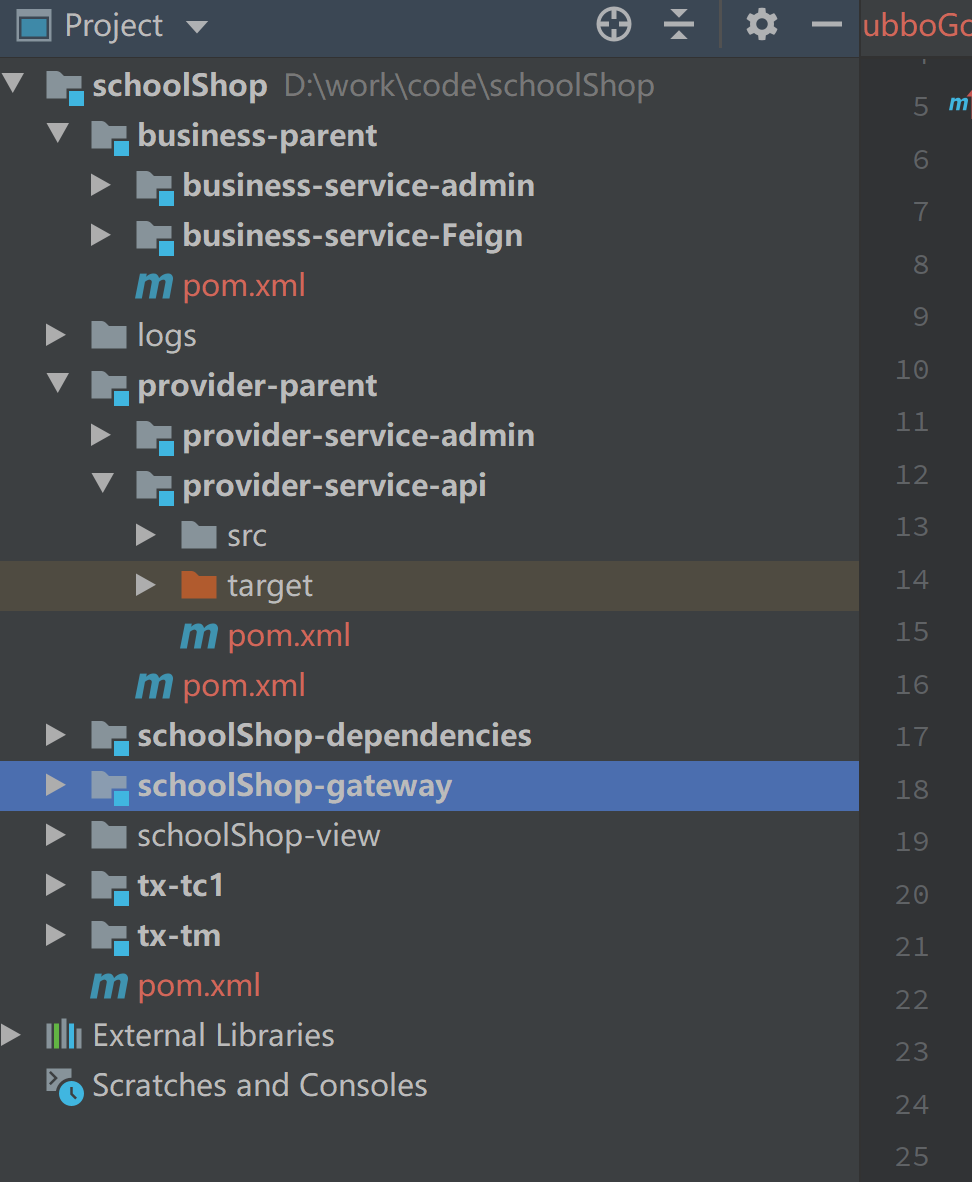


图3 项目主干结构

## 4.4 依赖导入

依赖导入是指在spring框架的基础上，通过maven插件，安装其他开发所需要 用到的jar包等别人已经开发设计好的框架或者工具，这些工具都是别人开发出来，经多人使用并且等到认可的，我们只需要把他从远程仓库下载下来，放在本地，即可使用，还能方便下次使用。

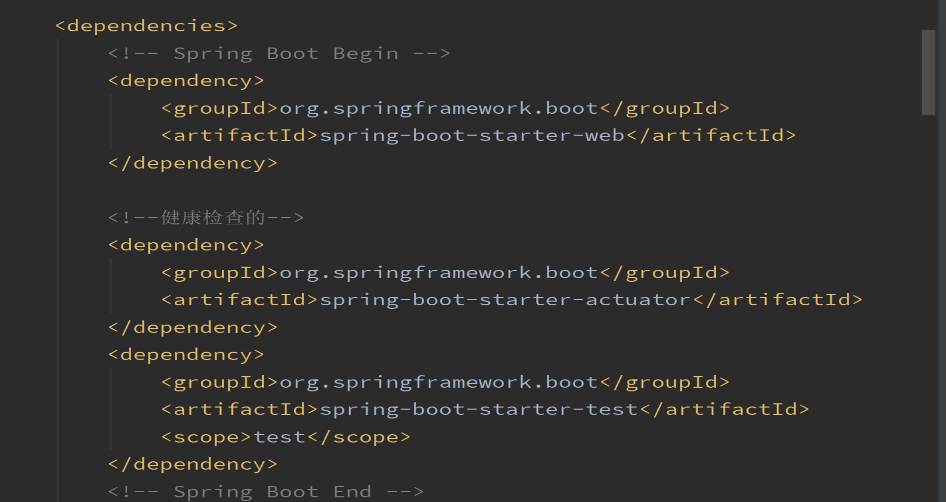


图4 依赖包导入案例

依赖包的导入例子如图4 依赖包导入案例所示

## 4.5 数据库连接

数据库连接使用了HikariCP（光），是日本公司编写的数据库连接技术，号称全球最快最好的数据库连接技术，所需依赖包如下：

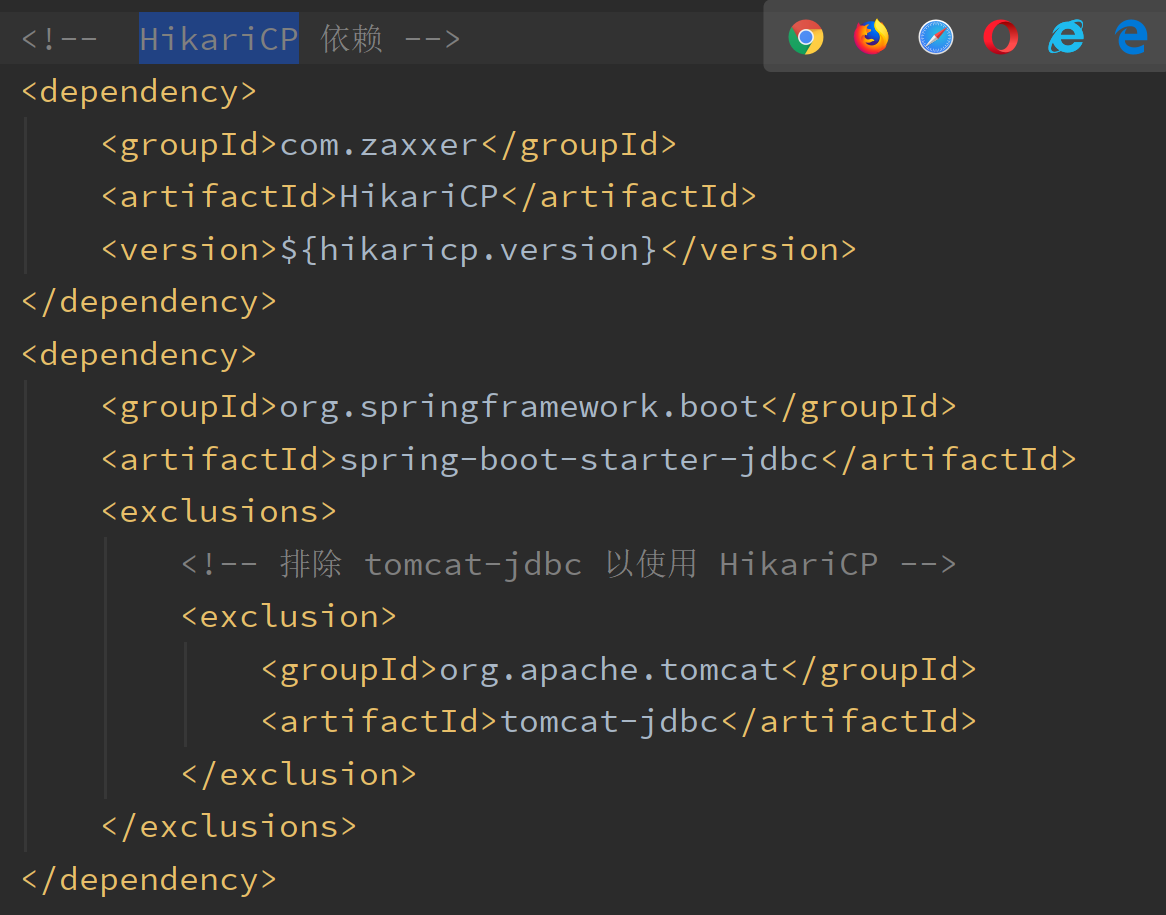


图5 数据库连接HikariCP

编写application.yml配置文件，在其里面，添加连接数据库的相关配置信息例如：连接数据库的名字、密码、数据库的IP地址和端口号、编码格式、最大连接超时时间等等；详情如下：

datasource:  
 type: com.zaxxer.hikari.HikariDataSource  
 driver-class-name: com.mysql.cj.jdbc.Driver  
 url: jdbc:mysql://XXX.XXX.XXX.XXX:3X06/school\_XXX\_shop?useUnicodeXXX=true&characterEncoding=XXX&useSSL=XXX&serverTimezone=Hongkong  
 username: root  
 password: root  
 hikari:  
 minimum-idle: 5  
 idle-timeout: 600000  
 maximum-pool-size: 10  
 auto-commit: true  
 pool-name: MyHikariCP  
 max-lifetime: 1800000  
 connection-timeout: 30000  
 connection-test-query: SELECT 1

## 4.6 目录结构

本项目使用spring Boot 开发标准，因此在使用idea开发工具创建项目时，选择spring boot项目创建，则会自动生成规范的目录路径，其中，文件入口为src目录，再到main目录，main目录下有一个Java文件和一个resources文件，可编译Java代码文件统一放在Java文件目录下，而resources为相关配置文件存放目录，所有文件必须严格按照要求放置指定的目录下，否则程序违法正常运行。

在java文件目录下，则是按照springMVC框架的开发标准构建的文件，分为controller、domian、service、mapper。其中，controller文件下的主要为被前端调用接口的Java文件，也就是三层架构中的控制层；domain文件夹里面主要是存放我们每一个数据库中的每一个数据表唯一对应的一个java实体类对象；service为业务逻辑层，里面包含着接口文件和接口对应的实现类文件，里面的接口主要提供给controller里面的方法调用，接口的实现类是最为主要的事务逻辑处理类，他不仅要对前端请求判断处理和审核，还要对作出的判断选择对应的执行流行，包括对数据库的具体操作，以及对操作结果进行判断处理，通过接口返回给调用方；mapper文件里面的主要包含被刚刚所说的service实现类所调用的接口。

再到resource文件，里面包含着上述的配置文件以外，还有着实现mapper文件下的接口实现类，这里的实现类主要的功能是根据所给的参数和需求，编写数据库语句，进行数据库操作，是三层架构里面的model层。

剩下的还有就是前端代码目录，里面包含着html、css、js格式的文件，以及所需的工具文件，组件文件。

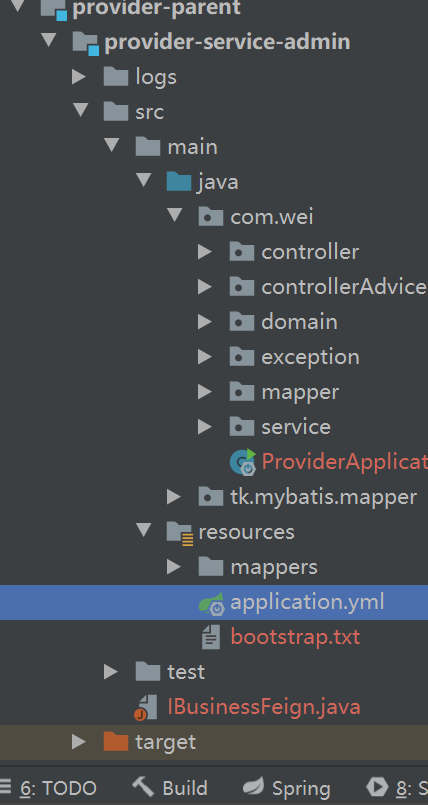


图6 目录结构

# 5 部分功能介绍以及代码实现

## 5.1登录页面

登录页面采用了html、cscs、js、jQuery、bookStrap、ajax为前端技术支持，其中，html为基础骨架，css为样式，js提供动态操作，bookStrap提供第三方组件库，使用ajax结合Jquery实现前端和后端的局部刷新页面和数据交互。

登录页面的功能点包括：用户名和密码输入，用户登录选项，管理员登录选项，以及跳转到注册页面按钮。前端提交代码如下：部分正则表达式，保证用户合法输入信息，避免sql注入：

$("#form").bootstrapValidator({  
 feedbackIcons:{  
 // 校验通过的输入框状态  
 valid: 'glyphicon glyphicon-ok',  
 // 校验失败的输入框状态  
 invalid: 'glyphicon glyphicon-remove',  
 // 校验中的状态  
 validating: 'glyphicon glyphicon-refresh'  
 },  
 fields:{  
 //用户名校验  
 username:{  
 validators:{  
 notEmpty:{  
 message:"用户名不能为空"  
 },  
 regexp:{  
 regexp:'^[\u4e00-\u9fa5]{0,}$',  
 message:"请输入中文名"  
 },  
 stringLength:{  
 min:2,  
 max:5,  
 message:"长度为2-5"  
 }  
 }  
 },

接着通过ajax技术提交数据实现局部动态验证用户信息以及融汇结果的校验：

var username = $("#Sno").val();  
var password = $("#password").val();  
//数据发送  
$.ajax({  
 url: "http://localhost:8080/secondhand\_war\_exploded/user/login",  
 data: {  
 username: username,  
 password: password  
 },  
 dataType: "JSON",  
 type: "POST",  
 xhrFields: {  
 withCredentials: true  
 },  
 success: function (resp) {  
 console.log(resp);  
 alert(resp);  
 //回调函数，结果判断  
 if (resp.status == 1000) {  
 alert(resp.data);  
 location.href = "index.html?uid=" + resp.data;  
 }  
 }

用户在输入用户名和密码等相关信息后，再选择对应的用户登录，或者管理员登录，登录信息提交到后端，获取前端传送的用户名和密码，调用对应的service接口里面的登录方法，在接口的实现类里面判断传送的用户名是否合法（非空等），如果合法，再从数据库中的用户信息表里面，进行用户输入的信息校验，校验成功，则返回该用户的用户id，否则返回登录失败信息；部分cotroller代码如下：

@Resource  
private IUserService userService;  
@RequestMapping(value="/loginXXXX",method=RequestMethod.POST,produces=MediaType.APPLICATION\_JSON\_UTF8\_VALUE)  
public String login( String name, String password){  
 String msg ="";  
 log.info("开始登录验证");  
 try{  
 msg = userService.logins(new User(name, password));  
 }catch (Exception e){

其中调用了userService中的登录接口，接口的实现类为：

@Override  
public String logins(User user) {  
 String msg = null;  
 log.info("service层");  
 //如果名字为空，抛出异常  
 if (user.getUserName().equals("")){  
 throw new loginException("用户名不能为空");  
 }  
 else {  
 msg = userMapper.logins(user);

userService接口的实现类则再把前端获取到的用户名和密码信息传送给dao 层（数据持久层），进行数据的最终校验；

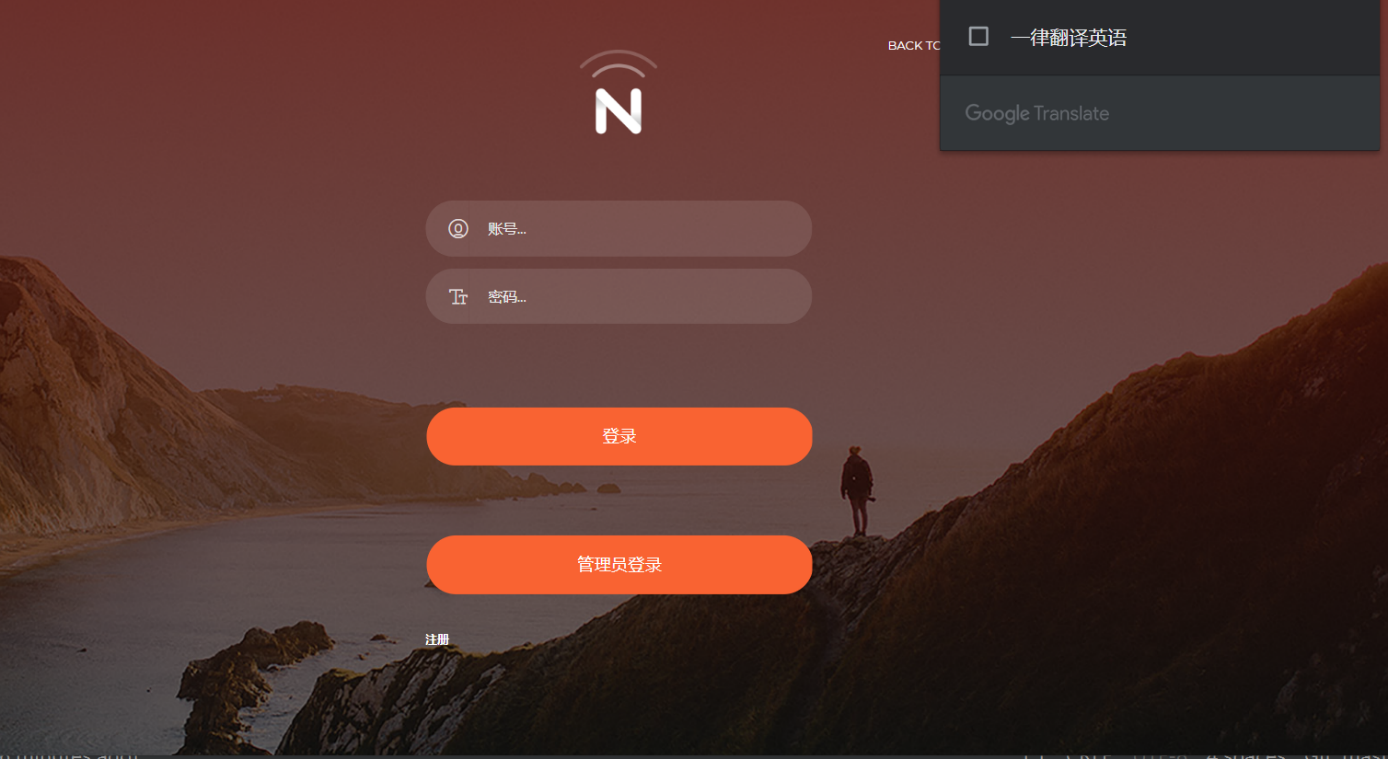


图7 登录界面

## 5.2用户注册界面

用户注册界面主要是把新用户输入的用户名、密码、邮箱、手机号等信息进行前端校验，然后把校验通过的信息传送给后端接口，后端接口再把信息交给业务层进行校验判断，校验加判断通过后，调用持久层，把用户信息添加到数据库中的用户信息表中，用户也可以返回登录界面直接登录已有的账号；其界面如下：

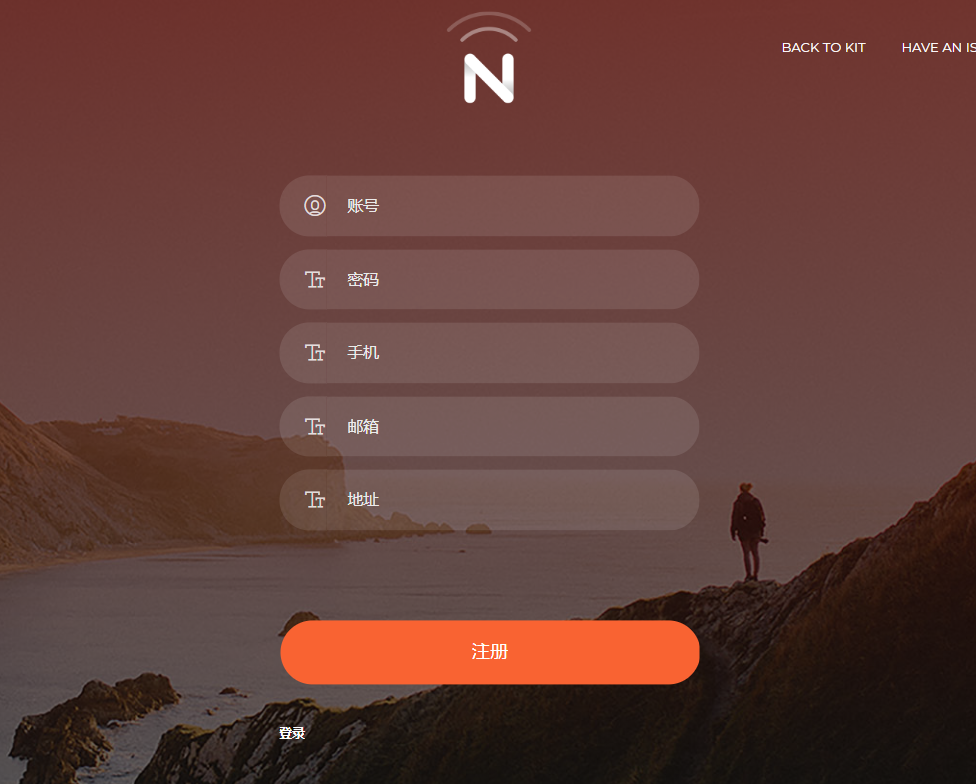


图8 注册界面

## 5.3首页展示

首页主要包含的功能有 （自上而下）：商品搜索、轮播图、个人信息中心（个人信息中心为收缩栏，打开里面包含着个人信息修改，个人上架的闲置物品管理，个人购买记录）、上传个人闲置物品功能、闲置物品分类功能等；

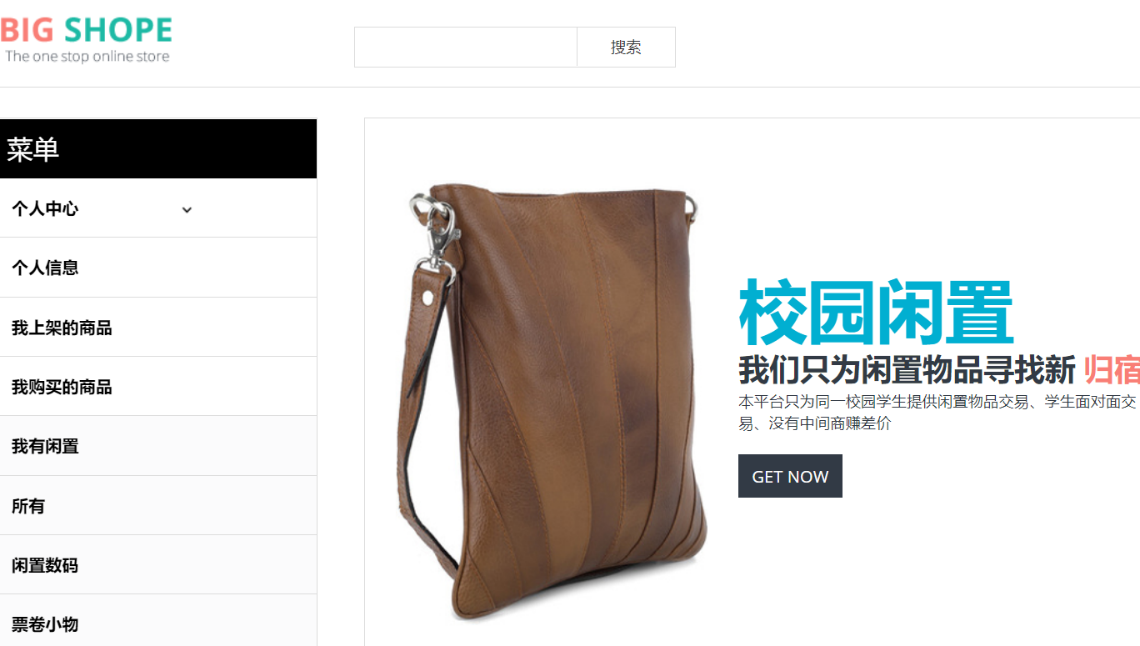


图9 首页

## 5.4个人信息修改功能

操作步骤：打开个人中心，展示出个人信息选项，点击个人信息，则弹出个人信息修改模态框，用户可在模态框在实现个人信息的修改，包括用户姓名、性别、爱好、电话、邮箱、地址的修改：

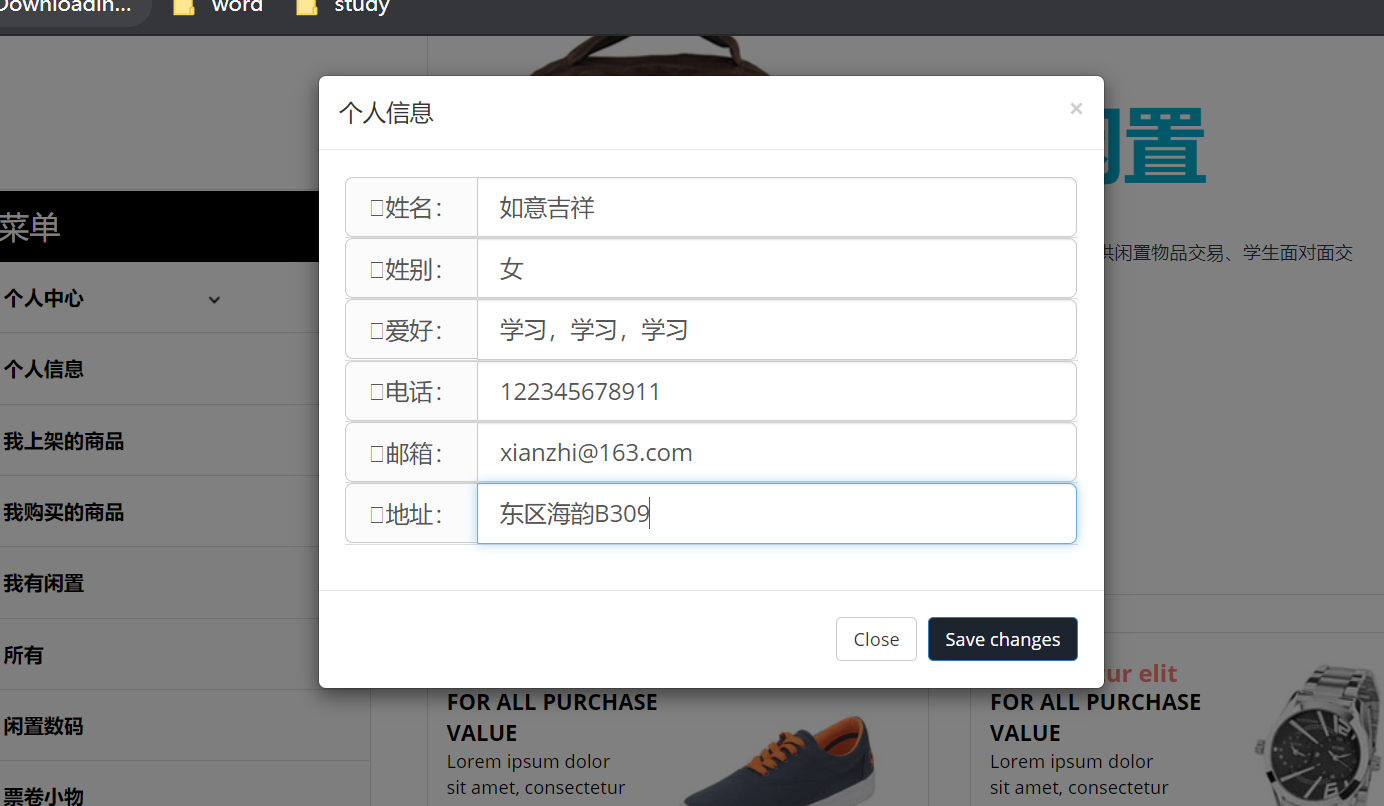


图10 个人信息修改

## 5.5上传个人闲置物品信息

上传个人闲置物品功能为用户提供填写闲置物品信息模态框，包括物品名字、物品原购买价格、物品现出售价格、物品简介以及上传物品照片。

其中上传照片成为开发时遇到的难题之一，涉及到前后端的io流传送，以及后端对上传照片的规范化命名和指定路径保存。具体操作如下：

在前端代码中，编写上传照片的模态框中添加一个form表单，form表单的恩仇type设置为multipart/form-data，当点击该改表达中的input时，自动打开本地设备的文件夹，用户选择自定的照片存放路径，勾选选择好的照片后，通过ajax局部刷新提交数据流到后台服务器中，前端部分代码如下：

<form id="form1" action="http://localhost:8080/admin/goodsController/myUpload"  
 method="post" enctype="multipart/form-data">  
 <input type="file" name="file" id="file"  
 style="background: url('asserts/img/上传.jpg') no-repeat;width: 500px;height: 500px;"/>  
 <input type="button" value="确认上传" class="btn btn-primary" id="myOnclick">  
</form>

后端服务器在controller层 接收到前端传送过来的数据流，参数格式为MultipartFile，在把数据流传送到service接口，通过sevice接口的实现类，调用了自定义的工具类，实现对上传的数据流进行格式规范化以及保存，部分工具类代码如下：

//获取文件名字  
 String originalFilename = file.getOriginalFilename();  
 //获取文件名后缀  
 String sufixName = originalFilename.substring(originalFilename.indexOf("."));  
 System.out.println(sufixName);  
 //重新命名  
 String newName= DateFormatUtils.format(new Date(),"yyyyMMddHHmmss")+sufixName;  
 System.out.println(newName);  
 //保存路径  
 File file1 = new File("F:/java每天的资料/second-hand-shop/views/assert-index/images/"+newName);  
 FileUtils.touch(file1);  
 //保存文件  
 FileUtils.copyInputStreamToFile(file.getInputStream(),file1);  
 return newName;  
}

最后返回规范化的照片名字给service实现类，实现类再把名字通过dao层存储到数据库，最终实现数据的存储。其中，照片的名字实现以当地实时时间精确到秒为照片名字前缀，照片格式统一规范为jpg格式；



图11 上传闲置物品

## 5.6 功能总结

本项目的功能根据设计理念和需求，主要为用户提供上传闲置物品信息功能和浏览闲置物品，以及获取联系卖家功能。由于经验不足，所能考虑到的功能不多，以及部分功能能还在开发跟新中，也处于设计的初衷，该平台只为在校学生分布和寻找闲置物品，处理同学们日常生活中的闲置物品，而非电商交易平台，所以并不支持在线支付功能。其他功能在日后使用反馈再进行分析和开发。

# 6 分布式服务注册和和管理测试

## 6.1 注册

本项目采用了spring cloud alibaba，使用的是阿里巴巴提供的注册中心nacos，注册中心的作用是把后端服务器统一注册，进行管理，只有注册过的服务，才能被第三方调用，不仅如此，nacos还为我们提供了配置中心，我们可以把注册文件上传到nacos的配置中心，这样可以实现配置文件的动态刷新，在修改了配置文件的内容后不需要重启服务。Nacos需要自己搭建，搭建结果如下：



图11 搭建nacos

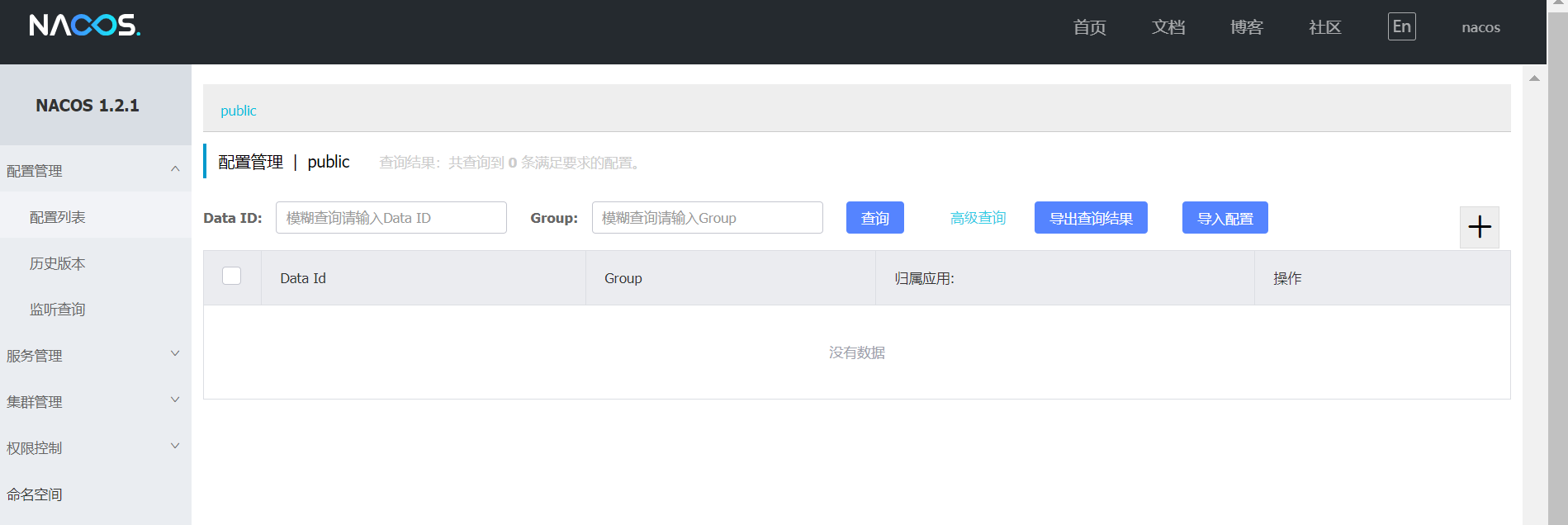


图12 nacos界面

## 6.2 流量守卫兵

本项目采用了阿里巴巴提供的流量守卫兵sentinel进行对请求的限流和设置反应时间、黑白名单设置、服务降级处理、授权管理等，可以实时跟踪请求的响应时间和服务的健康状态。

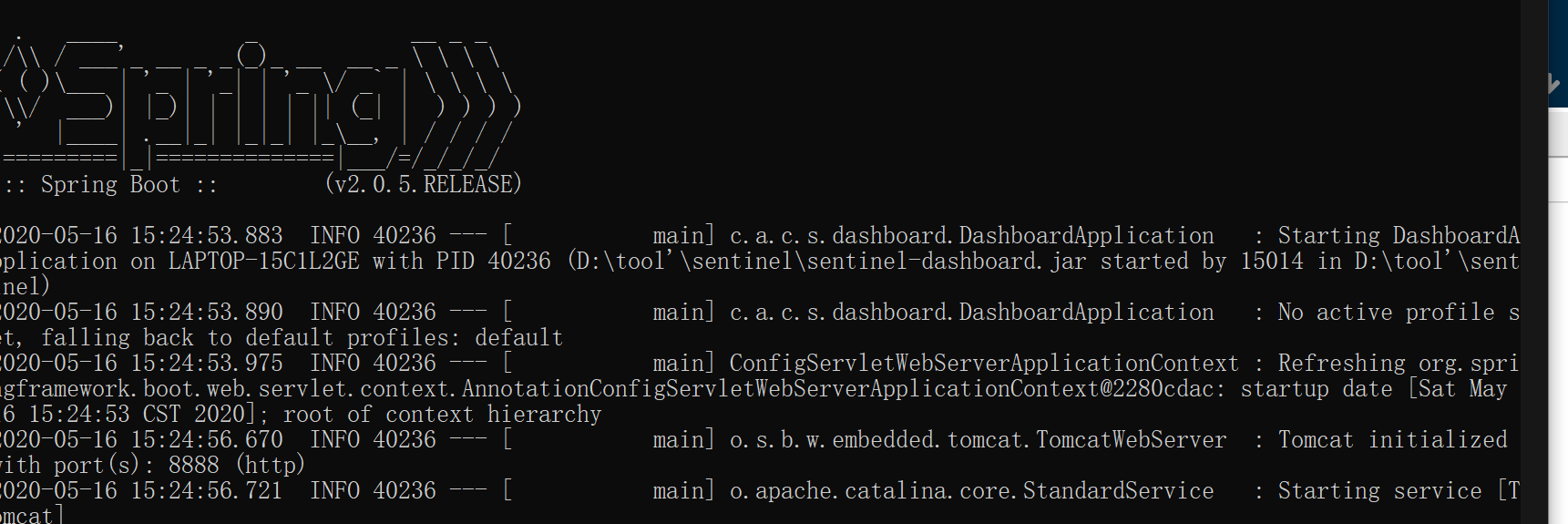


图13 搭建sentinel



图14 sentinel登录界面

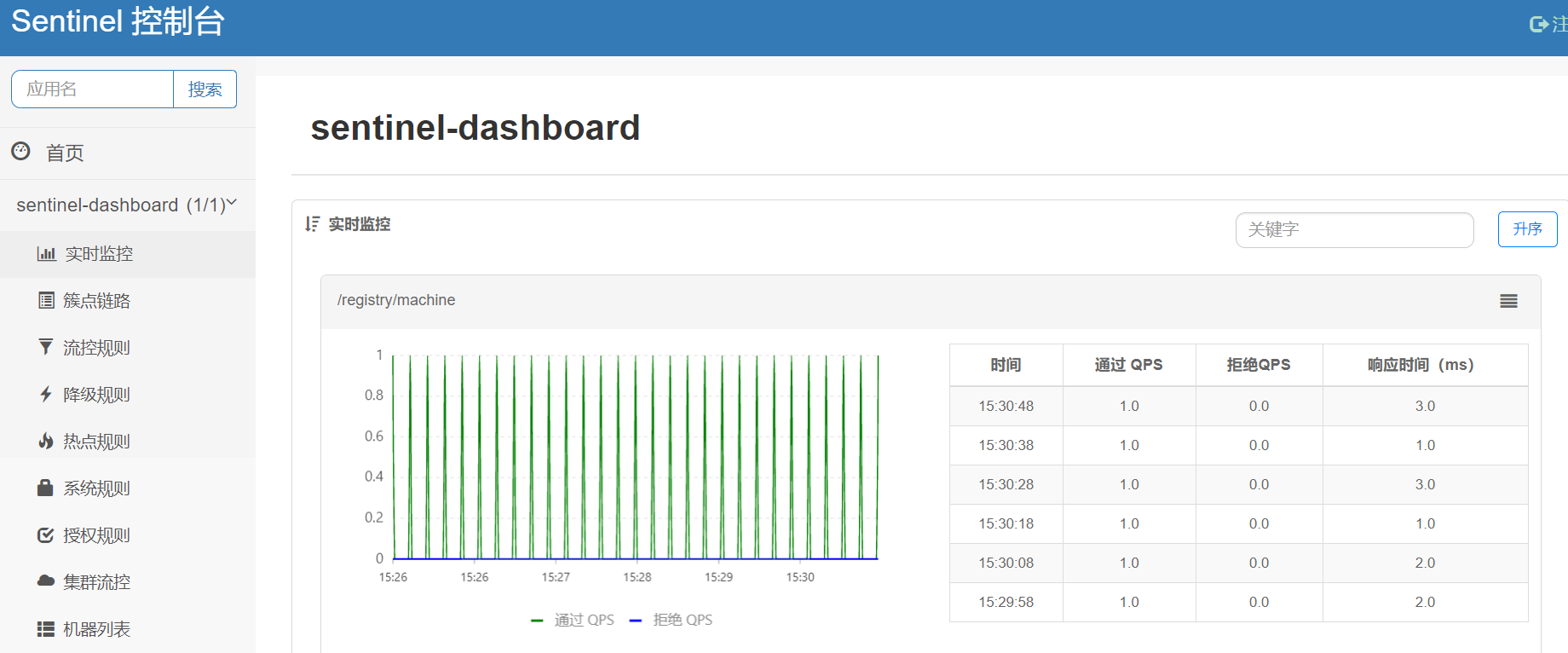


图15 sentinel服务测试



图16 sentinel服务测试结果

测试结果及分析：

由于还没有实际的产品用户量，因此使用模拟用户请求进行了并发量测试，测试结果入上图，测试反馈结果良好，达到预期目标，每个请求的响应时间均在3~5毫秒之间。

产品部署：

对于产品的部署上线，购买了一个1核2G的阿里云服务器，使用Docker容器化引擎进行对每个服务的搭建和运行，但由于服务器出现了故障，导致产品的上线时间延误，无法按期完成。

# 7 项目总结

从该产品的设计理念开始，到产品的最终部署上线，中间涉及的知识范围包含着需求分析、前端UI设计、前端页面实现、后端接口编写（关于分布式系统开发涉及到的知识点较多，前面以有介绍）、以及对服务器的操作（Linux、docker[12]、docker-compose）和运维（Linux系统的环境搭配等）、产品的测试和上线等，在此感受到编程语言的强大。

由于经验的不足以及开发周期较短，该项目仅仅是个小型项目，究其到底其实并不能很好的突出微服务的优势的，按照如今互联网的发展速度，传统的单体应用必将不能满足不断庞大的用户并发量，分布式系统开发的优势却随着用户量的增加而日显强大，市场对分布式系统开发人员的需求也相当大。该项目按照了分布式系统开发的开发思想，对功能的拆分，以及对每个服务之间的通信和管理，架构的选择都是目前市场流行的方向，尽管产品存在不少的不足，尚未能正真创造价值，但通过对该产品的设计和开发，本人感到受益极为丰富，对各种之前从未认识的知识的学习和研究，通过网络学习和线下培训，很多知识点从无到有，让我开阔了视野，了解到市场的工作技术需求，并且掌握了真正能够投入开发的编程技术，对日后就业起到绝对性的帮助。

# 鸣 谢

由衷感谢广东海洋大学在入学至今四年来给予我的培育，在校期间，不仅让我学习到了许多课本上的知识，还带领我们见识到了许多成功人士的成长历程，使我们可以借鉴学习和成长。感谢学校为我们提供的良好学习资源和学习环境以及丰富的校园生活，让我们拥有步入社会的知识和技能，为日后的工作做好了铺垫。

感谢各位良师益友，在你们的帮助和教育下，学习了许许多多新的编程知识，你们的指导和鼓励是我的毕设能够顺利进行的基础磐石 。

感谢大学四年间的同学和舍友，营造了一个良好的学习氛围和环境，使我们共同努力和进步，获得扎实的知识，为步入社会做好充足的准备。

最后特别感谢各位老师和导师，为我指明了前进的方向，授予我丰富的知识和工作技能，以及指导和帮助我解决毕业设计所遇到的问题，使我能更好地完成毕业设计。

# 参考文献

1. 龚鹏，微服务分布式架构开发实战[M]，人民邮电出版社，2017.
2. 陈雄华，Spring 3.X企业应用开发实战[Z]，电子工业出版社，2012
3. 方志鹏，深入了解spring cloud与微服务构建[J]. 人民邮电出版社，2018,3:3-92
4. 马雄. 基于微服务架构的系统设计和开发[J]. 北京邮电大学，2017,1:18-213
5. 王国辉．Java Web编程宝典:十年典藏版[M]．北京：人民邮电出版社，2011．
6. 李刚．轻量级J2EE企业应用实战[M]．北京：电子工业出版社，2007.
7. 杨静．基于JAVA WEB中MVC模式的研究与应用[J]．电脑知识与技术，2014,1:45-118
8. 李孟燃．网页设计中计算机多媒体技术的应用[J]．计算机光盘软件与应用，2014,1:45-218
9. David Schultz，Beginning HTML with CSS and XHTM[Z] ，人民邮电出版社，2008.
10. 阎新芳．Java设计模式在数据库编程中的应用研究[J]．信息通信，2014,1:15-128
11. James Turnbull，The Docker Book[Z] , 中国工信出版集团，2011.
12. Christopher Negus，Docker Container[Z]，中国工信出版集团，2012.