

摘要

随着科学技术的不断发展，人们的生活品质也在不断地提升，足不出户的线上购物方式就是一种很好的体现。越来越多的人选择在网络平台上购物，这种购物方式不仅款式多样，价格比实体商店便宜，更重要的是非常便利，只需要一部手机就可以完成购买，而且操作方便，即使没有经验的人操作一遍之后也就会明白了。但是，线上购物带给我们便利的同时也会有许多的弊端，比如造成资源的浪费，人们往往会因为冲动消费而购买一些商品，但是买回来却发现不合适或者后悔，这样商品的价值便得不到提现。还有一些人因为资金问题可能会把商品变现，解决燃眉之急。所以，二手物品交易网站的存在是非常有必要的，帮助卖家解决商品闲置问题以及买家的性价比需求。本系统是基于市场调查的结果，实现了人们买卖二手商品的交易需求。

关键词：二手物品交易 JAVA 商品 服务器 前端框架

Abstract

With the continuous development of science and technology, people's quality of life is also constantly improving. Online shopping without leaving home is a good embodiment. More and more people choose to shop on the network platform. This kind of shopping method is not only diverse in style and cheaper than the physical store, but also very convenient. It only needs a mobile phone to complete the purchase, and the operation is convenient. Even if the inexperienced people operate it again, they will understand it. However, while online shopping brings us convenience, it also has many disadvantages, such as waste of resources, people often buy some goods because of impulse consumption, but they find it inappropriate or regret when they buy them back, so the value of the goods can not be raised. There are also some people who may cash in goods because of capital problems and solve the urgent problem. Therefore, the existence of second-hand goods trading website is very necessary to help the seller to solve the problem of idle goods and the buyer's cost-effective demand. This system is based on the results of market research, and realizes the transaction demand of people buying and selling second-hand goods.

Key words : Second hand goods trading JAVA commodity The
server Front end frame

目 录

第一章 绪论	1
1.1 课题背景	1
1.2 课题的研究目的和意义	1
1.3 国内外研究现状	2
1.3.1 国内研究现状.....	2
1.3.2 国外研究现状.....	3
1.4 研究内容	4
1.5 使用的技术简介	4
1.5.1B/S 结构模式.....	4
1.5.2JSP 技术.....	5
1.5.3Mybatis 技术.....	6
1.5.4Tomcat 服务器技术.....	7
1.5.5JQuery 技术.....	7
1.5.6Bootstrap 框架.....	8
第二章 需求分析	9
2.1 系统功能分析	9
2.2 系统性能需求	12
2.2.1 准确性和及时性.....	12
2.2.2 易用性.....	12
2.2.3 安全性.....	12
2.3 可行性分析	12
2.3.1 技术可行性.....	12
2.3.2 操作可行性.....	13
2.3.3 设计可行性.....	13
2.3.4 经济可行性.....	13
第三章 系统设计	14
3.1 系统架构	14
3.2 系统功能模块设计	14
3.3 系统工作流程设计	16
3.4 数据库设计.....	16
3.4.1 概念结构设计.....	16
3.4.2 逻辑设计结构.....	17
3.5 关键代码	20

第四章 实现效果	30
4.1 用户界面	30
4.1.1 登录注册.....	30
4.1.2 商品发布与查看.....	31
4.1.3 个人主页.....	32
4.1.4 钱包管理.....	34
4.2 管理员界面	34
第五章 系统测试	37
5.1 测试的目的与目标	37
5.2 测试方法	37
5.3 测试用例	38
5.4 测试总结	39
第六章 总结与展望	40
第七章 致谢	41
参考文献	42

第一章 绪论

1.1 课题背景

“二手闲置物品”对于人们来说，处理总是一个难题。用不到的东西放着不仅浪费，而且占用空间，丢掉的话又会觉得可惜。二手闲置物品的分类也是各种各样，比如生活用品、电子设备、电器等都是二手闲置物品市场中普遍存在的东西，由于人们的日常用品具有较强的重复使用特性和自身的不稳定因素，使得二手市场存在着更稳定的需求，这种二手市场的存在增加了资源的利用率，值得大家鼓励和认同。但是由于二手市场的运行体制存在较大的漏洞，因此应该建立一套科学合理的治理体制来实现规范管理，进而促进二手市场这类资源的合理配置，同时提高商品的利用率。综上所述，我觉得开发一个二手闲置商品交易网站是非常有意义的，也符合我们当下年轻人的生活态度。

1.2 课题的研究目的和意义

目前，我国正处于互联网发展的高速阶段，几乎每家每户都需要互联网来进行工作，例如学生上网课、在线看电影等，网络的身影无处不在。在电子商务方面，网络的发展改变了我们传统的购物方式，我们从过去的线下购物慢慢地转变为线上购物，人们享受着足不出户的购物体验。如今大火的直播带货，就是电子商务成功的典范。所以网上购物成为许多人的第一选择。凡事都有两面性，线上购物也同样如此，虽然它十分方便快捷，但也带来的一些不好的地方。研究发现，随着生活品质的提高，人们的购买力也随之增高，产品更新速度也随之加快，每个人都会因为冲动消费而闲置了许多二手商品，比如衣服、电子产品、生活用品等。但是，对于这些闲置物品的处理，许多的人因为没有有效的处理途径，而选择丢弃或者放着，造成了大量不必要的浪费，无法让其发挥出价值。

所以，我决定开发一个二手闲置商品交易网站，为那些人们不需要用到但仍然有价值的商品提供一个出售的平台。不仅仅是让人们有机会处理二手物品，还让那些经济能力不太强的学生有机会以更低的价格买到他们想要的东西。开发二手闲置物品交易平台，有利于培养人们的低碳意识和社会责任感，不仅可以具有客观的经济效益，更重要的是可以让人们认识到节约资源的重要性。

1.3 国内外研究现状

1.3.1 国内研究现状

如今，越来越多的国内公司企业已经明白了发展电子商务的必要性，并且已经开始行动起来。从阿里巴巴的淘宝，到如今的直播卖货，越来越多的企业在电子商务中找到了商机，特别是今年的疫情原因，做线上电子商务的更是迎来了大爆发，相比于线下的商店，电子商务更能够适应许多突发的情况，从而避免损失。目前我国的电子商务主要分为以下几种模式：

- 1) B2B 模式：即商家对商家的电子商务模式。简而言之 b2b 就是企业与企业之间营销活动的平台，是现代互联网市场领域的一种商业形式。我国典型的 B2B 企业就是阿里巴巴，它通过企业内部特定的网络来完成交易。
- 2) B2C 模式：我国最早出现的一种电子商务模式。简单来说，B2C 就是一种企业对个人的线上购物模式，在我国 8848 就是首个开创 B2C 模式的企业。如今，淘宝、京东等企业的出现，使得 B2C 模式已经成为了电子商务的主流模式，从十几岁的小学生，到六七十岁的老年人，只需要一部手机，就可以体验到足不出户的购物体验，对于上班族来说更是一个完美的选择。
- 3) C2C 模式：众所周知，C 的英文单词是 consumer，即消费者。所以，C2C 模式指的就是两个消费者之间相互交易的一种电子商务模式。通常用于商品拍卖、转让等方面。本系统所开发的二手闲置商品交易市场就是基于 B2C 的一种电子商务模式，由消费者出售自己的商品，另一个消费者购入商品。
- 4) C2B 模式：指的是消费者与企业的一种电子商务模式，这对于电子商务来说，是一种全新的模式。目前我国使用 B2C 模式的企业有摇篮网、宝宝树、豆瓣网等。他们首先在平台上建立社区，吸引人们的加入，等到社区的人到达一定程度的时候，他们就会利用人数的优势，让人们通过批发的价格购买商品，从而获取利润。

根据 ACCS 的调查，通过比较中国内地和香港的信息化进程的实际情况，最终得出的结论是，最大的区别是“电子商务”。虽然，我国已经开始意识到电子商务的重要性，但是在实际生活运用中并没有充分考虑社会需要和环境的特殊性，因此，我们的电子商务主要还存在以下几个问题：

- 1) 安全性较差，我们都知道，如今的网络世界根本没有秘密可言，基于大

数据的网络平台更是可以通过我们自身的喜欢，来推销产品。这样给人

的一种感觉就是没有安全保障，就好像自己所有的东西都能被别人知道。比如我前几天下载了一个软件，仅仅用了他们的微信授权登录功能，过几天他们就打电话过来了，这给我的感觉就是十分的不安全。

- 2) 相关的法律法规文件尚未完整。在网络上，有许多的就是利用法律的漏洞，来获取利益。比如近几年，有一些人在网络上散布广告，可以免费赠送商品，只需要出运费就可以了，可是许多人给了运费之后，却迟迟没有收到货，意识到的时候自己已经上当受骗了，知道后来广告法颁布之后这个问题才得到了缓解。类似这样的问题还有很多，我们需要更完善的法律机制，才可以保障到电子商务的稳定发展。
- 3) 搜索功能的不完善，我们在平时的购物软件中，随便输入一个关键词，便可以搜索出许许多多的商品，但我们的一个难题就是，有这么多的购物网站，有这么多的商品，我们如何从中选出性价比最高的商品呢？

1.3.2 国外研究现状

近年来，其他国家的电子商务也在迅速发展，网上支付与网上购物系统等都较为健全，给人们的生活带来了方便。举例来说，1994年，杰夫·巴佐斯（Jeff bezos）放弃了在纽约市的投资公司的工作，并在他位于西雅图市郊的租来的车库里创办了amazon.com，这是美国第一家在线零售公司。高达\$ 300000的资金。在公司成立初期，只有三个“sun”工作站和300个免费试用客户。一开始的时候，贝索斯用了仅仅一年的时间就开发出了亚马逊平台以及存放数据的数据库。1997年，亚马逊便在美国成功上市，那时候的股价是每股18美元，这在当时已经展示出了其巨大的潜力。截至交易日结束时，亚马逊的股票已被拆分三倍，股价更是飙升到了每股94美元。经过几次扩容后，亚马逊的股价已经稳定在了每股89美元。经过了四年的发展后，1999年，亚马逊的市值达到280亿美元，远远超过拥有113年历史的零售公司Sears的158亿美元，并且远远超过Barnes & Noble的市值（13.5亿美元）。自1994年成立以来，亚马逊已经超了几乎所有的同行公司，成为全世界最大的“虚拟超市”。亚马逊使得其他（Barnes & noble）也依靠互联网销售商品的公司，几乎每天都生活在梦想中。仅仅5年时间，亚马逊就利用其庞大的客户信息数据库，以及安利等密集的网络服务系统，几乎挤进了所有的岗位，并获得了微软等公司的“垄断地位”。

如今，amazon.com 已经成为全球第三大图书网站。该公司有来自 160 多个国家的 1300 万长期客户。网上书籍的日销售量达到 57000 本，销售额超过 199 亿美元。现在亚马逊的业务范围已经扩展到音像、礼品、玩具、拍卖品等等。由于善于利用网上购物的优势和灵活的营销方式，亚马逊得以获得如此巨大的成功。它具有广泛的信息资源和书籍选择，因此无需中间商就可以大大降低成本。另外，购买过程很简单，减少了人的消费，节省了消费者的时间，等等。在传统书店里很难做到。

截止到目前，大部分的国家 and 地区对于电子商务的发展都十分重视起来，他们也在逐渐开创越来越多的电子商务公司，然而，问题也会随之而来。由于大部分公司未对人们是否熟悉计算机这个因素实行充分的思考，导致操作繁琐，系统的易用性还需要提高，并且就安全支付层面，亦具有许多的问题，还有待提高。

1.4 研究内容

本文的研究内容包括用户管理、商品管理、后台管理、用户操作界面、商品发布与删除、收藏商品、商品搜索以及购买等功能模块的设计。主要介绍二手闲置物品交易网站的设计分析和开发的整个过程，并对系统的各个模块加以详细说明。本课题研究采用了软件开发的思想和，基于多层次的系统结构开发。

主要内容：

系统的需求分析：针对系统的目标、需求分析进行详细的分析与模块划分。

每个功能模块设计：这主要指的是系统从界面到核心功能、代码的设计。

1.5 使用的技术简介

1.5.1B/S 结构模式

B/S 结构，即浏览器和服务器结构。在互联网飞速发展的今天，人们上网的方式也发生了变化，越来越多的人选择更简便的上网方式。而 B/S 就是在这种情况下诞生了。它属于一次性开发，实现了不同的人群可以通过他们自己的方式去访问互联网以及操作数据库，例如每个人用的设备都不一样，每个人用的浏览器都不一样，这些都是可以选择的。这样的操作方式不仅可以很好地保护到服务器端数据的安全，也可以方便开发者管理用户访问的权限。

B/S 模式的最大优势就是它在何时何地都可以运行，不像 C/S 模式那样需要安装特定的软件。它不需要我们安装任何特殊的软件，只要我们电脑能够上网，有网络就可以了。因此，B/S 结构的客户机维护成本可以说是零成本，而且对于系统的维护和扩展也非常地方便。B/S 的特点就是分布式，在使用方面，可以随

时随地处理业务，访问的方式既简单又方便，如果我们需要更多的功能或者需求，只需要在浏览器中添加插件或者网页即可。在维护方面，开发者只需对页面进行修改，就可以使所有用户都获得同步的更新，维护方便，共享性强。

B/S 结构模式的原理如图 1-1 所示：

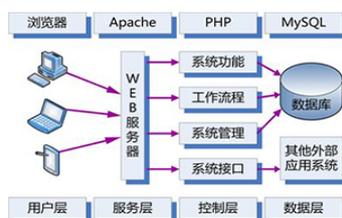


图 1-1 B/S 模式原理

与 C/S 结构相比，B/S 结构基于浏览器，跟操作系统没有密切关系。因此，跨平台是 B/S 结构的独特优势。由于投入成本，硬件投资保护等较低，系统的维护和升级成本较少，B/S 结构被越来越多软件开发人员追捧。

1.5.2 JSP 技术

Jsp 是由 Sun 公司提供的动态 web 开发技术。我们通常使用 JSP 来开发的页面或者软件都是基于 Java 的。通过 JSP，我们可以轻松实现动态页面，根据利用 jsp 提供的 api，我们可以获取到后台传过来的数据，并把它显示到页面上。当数据发生改变时，我们的页面也会随之改变，做到及时更新。Jsp 的优点如下：

- 1) JSP 程序基于 Java 语言，因此它可以使用各种功能强大的 API，如 JDBC、jaxp、JNDI、EJB 等。
- 2) JSP 可以与 Servlet 相互配合使用，提高开发的效率。使用 Servlet 来实现后台逻辑的处理，而 JSP 便可以更专注于前端和后端的交互。
- 3) 使用 jsp 技术，我们就可以省下了许多写前端代码的时间，例如我们平时如果需要获取接口的话，以 vue 框架为例，我们就需要通过 axios 来获取 api，再把数据保存到 data 中，然后再在页面中渲染出来。而 JSP 便可以一步到位，我们知道利用它提供的 api，在后台返回的 model 中获取到数据，直接在页面中渲染出来，节省了我们写 js 代码的时间。

1.5.2.1 Spring 技术

Spring 技术的诞生就是由于我们在软件开发生的过程中，代码越来越多，使得代码的后期维护和管理会变得非常麻烦。当我们有了 Spring 技术，开发过程就会被简化，我们可以把对象交给 Spring 容器来管理，例如 Bean、Dao、service 等，这样就会大大提高我们开发效率，同时也更方便代码的后期的维护与管理。Spring 技术的优点是：

- 1) 使用 spring，用户不再需要为非常低级的需求（例如单实例模式类，属性文件解析等）编写代码，而可以将精力更多地集中在上层应用程序上。
- 2) Spring 为我们提供了 AOP 功能。AOP 可以轻松处理许多传统 OOP 难以实现的功能。
- 3) 在 Spring 中，我们可以通过声明的方式管理事务，摆脱繁琐的事务管理代码，提高代码的质量和开发的效率。
- 4) Spring 可以减少使用各种框架的难度。Spring 为各种出色的框架（例如 struts, hibernate, Hessian, quartz）提供了直接支持。

1.5.2.2 SpringMVC 技术

为 Spring 模块提供了 MVC 模型，将 web 层进行职责解耦，基于请求驱动指的就是使用请求-响应模型，框架的目的就是帮助我们简化开发。

SpringMVC 运行原理如图所示：

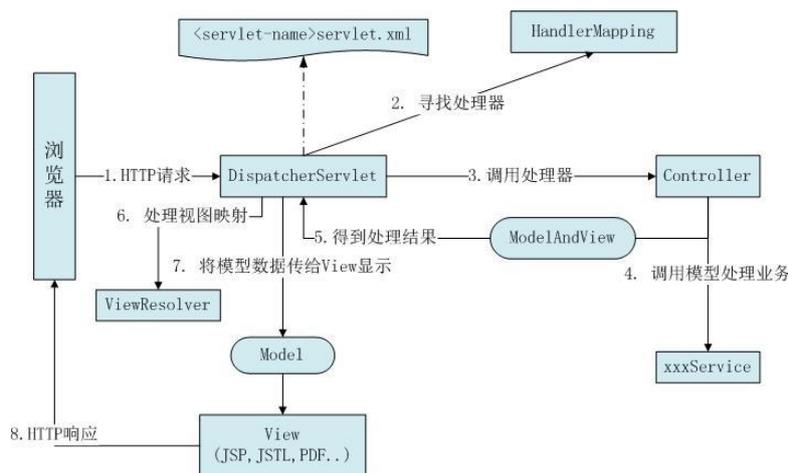


图 1-2 SpringMVC 运行原理

1.5.3 Mybatis 技术

Mybatis 基于 Java 的优秀持久层开源框架。使用 Mybatis 能够自主选择

SQL 语句，将数据库连接操作封装起来，其存储过程也极为方便。MyBatis 几乎去除了全部的数据库连接代码，省去手动设置参数和检索结果。除此之外，使用 XML 映射文件或者注解，都可以通过接口和类映射成为数据库中的记录。

Mybatis 的优点有以下 3 个：

- 1) 与 JDBC 比较，减少了超过一半的代码量。
- 2) 容易上手，且可重用。
- 3) 低耦合，便于统一管理和优化。

1.5.4 Tomcat 服务器技术

Tomcat 是一个免费的 web 应用程序服务器，在没有阿里云或其他线上服务器的时候，Tomcat 就为我们本地测试提供了一个很好的环境。Tomcat 是由 Apache 和 sun 公司共同开发出来的产品，目的就是为了给我们提供一个调试 jsp 程序的环境。Tomcat 经过的多年的历史发展，目前已经有了 9.0 的最新版本，本系统就是一个基于 Tomcat9.0 的二手闲置商品交易网站。

1.5.5 JQuery 技术

jQuery 是一个轻量级 JavaScript 框架。它最大的优势就是可以兼容浏览器的版本例如 IE8 等，不需要我们再去针对不同的浏览器去写兼容代码，这就是我选择 jQuery 框架的原因。只要我们使用了 jQuery，我们就可以通过写更少的代码，实现更多的功能，这也是 jQuery 框架的宗旨，在我看来，它已经做到了。

jQuery 的主要优点有：

- 1) 它是一个非常简洁的 js 库，没有其他炫酷的特效，但却可以帮助我们解决许多的日常开发问题，实现许多的功能。如今，它已成为我前端开发工作比用的框架之一。
- 2) 在 jQuery 的官网上，有非常详细的文档说明，我们只需要查阅文档，找到我们需要实现功能的代码，在对照着写下来就可以了，非常地方便。
- 3) 通过 jQuery，我们可以随时随地从我们的 html 代码中获取到需要的元素，进而可以使得 js 代码与 html 代码分离，保持代码的整洁，也更方便我们调用每一个元素。

1.5.6 Bootstrap 框架

Bootstrap 是一款开源的基于 HTML、CSS、JavaScript 的前端框架，它最大的好处就是支持响应式开发。通过 bootstrap 我们就不需要针对不同的设备来写好几个页面了，我们只需要在一个页面通过媒体查询针对不同的设备显示不同的效果就可以了，这是 bootstrap 最吸引我们程序员的地方。虽然现在 layui 更加地流行，但是我还是习惯 bootstrap 框架，因为在我实习的公司，就是使用这个框架，使用起来更加地得心应手。不仅如此，bootstrap 还提供了许多的 css 页面美化样式和许多的插件，例如本系统中上传图片的插件就是使用了 bootstrap 的。

第二章 需求分析

2.1 系统功能分析

二手闲置物品交易网站是根据各大二手平台的实际情况按照网络化的需求开发的，它的目标非常明确，即通过建立二手网络交易平台，将原来二手商品线下交易方式转变成通过网络平台进行交易的线上模式，使二手交易更加便捷、安全、规范和有针对性。

系统功能需求整体上分为用户使用功能和系统管理功能，其中用户使用功能包括用户浏览商品、用户登录、用户注册、用户搜索商品、用户发布商品、用户支付、用户查看个人信息等功能；系统管理功能包括管理员管理用户、管理员管理商品、管理员管理订单、管理员管理用户钱包和管理员管理系统设置等功能。

根据以上功能需求分析，通过用例图来描述系统的主要功能。构建用例模型的第一步是确定模型中的使用者有哪些，确定使用者的原则有：谁是系统的维护者、谁是系统的参与者等。维护者处于系统内部，对系统有绝对的控制权；而参与者一般都位于系统的外部，处于系统的控制之外。

现在确定本系统用例模型有三种，分别是游客、注册用户和系统管理员。下面分别对这三个角色的功能进行描述：

1) 游客

游客是未注册的用户，他们可以浏览商品，可以搜索商品，如需购买商品，必须先注册成为网站用户。游客主要功能如下（图 2-1 为游客用例图）：

- a. 浏览商品
- b. 搜索商品；
- c. 注册成为网站用户。

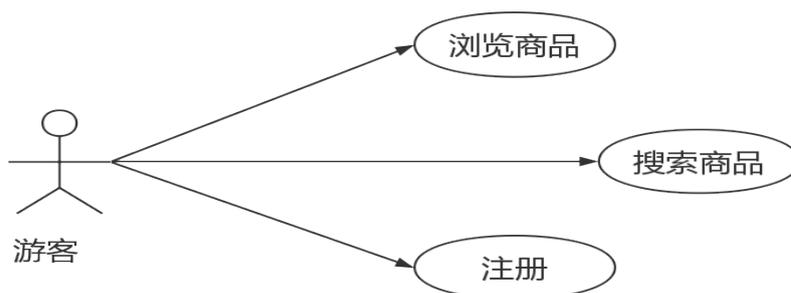


图 2-1 游客

1) 注册用户

注册用户是经过网站合法认证的用户，登录网站后可以浏览商品、搜索商品、发布商品、关注商品、购买商品和查看个人中心。用户注册后的主要功能如下（图 2-2 为注册用户图）：

- a. 浏览商品；
- b. 搜索商品；
- c. 关注商品；
- d. 发布商品；
- e. 购买商品；
- f. 查看个人中心；
- g. 发布求购信息；
- h. 登录网站

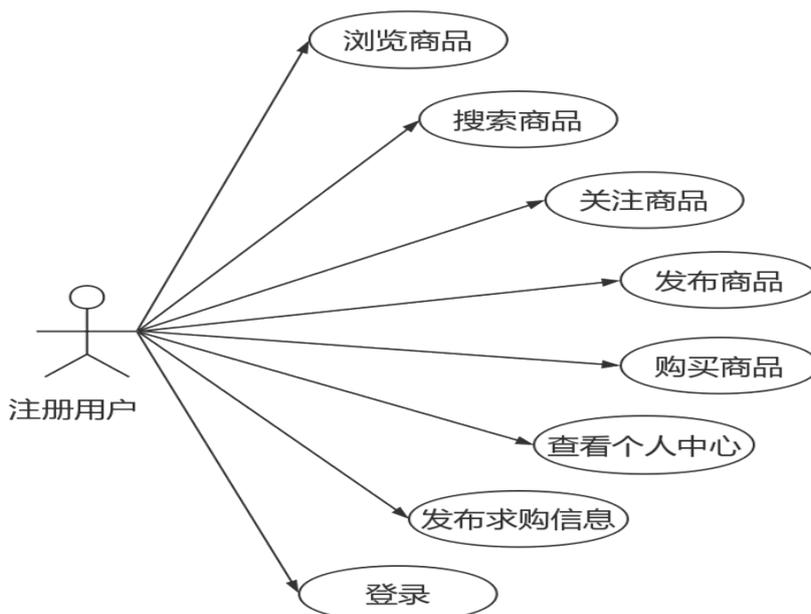


图 2-2 注册用户

2) 系统管理员

系统管理员主要负责系统的后台管理工作，主要功能如下（图 2-3 为系统管理员图）：

- a. 管理用户；
- b. 管理商品；
- c. 管理订单
- d. 管理钱包；
- e. 管理系统设置；

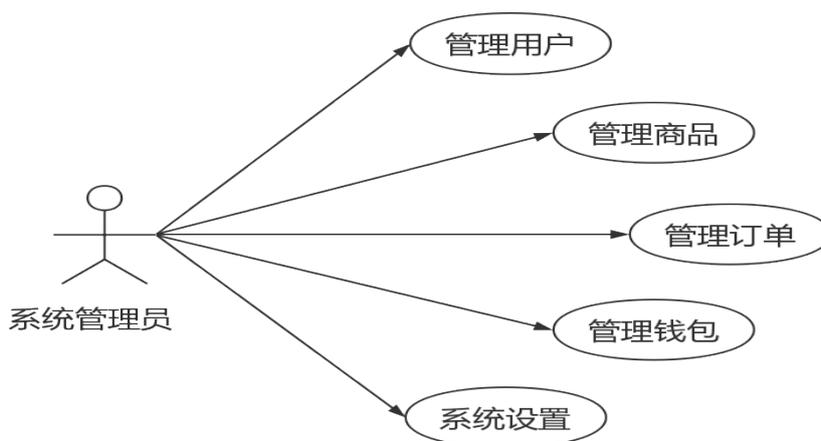


图 2-3 系统管理员

在确定了系统用户和用户功能后就可以构建二手闲置物品交易网站的用例图了，整个系统的图 2-4 系统总体图所示：

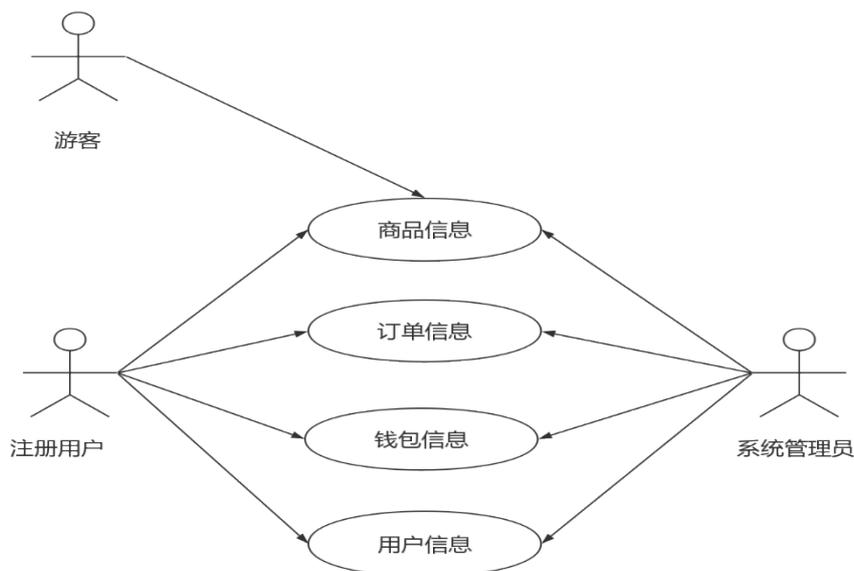


图 2-4 系统总体

2.2 系统性能需求

2.2.1 准确性和及时性

系统应具备能够准确无误、及时响应用户的操作的性能。由于用户的自主操作是整个系统的主要组成部分，所以准确性、及时性对整个系统有特别重要的影响，因此有必要对系统进行多次的测试。对于用户而言，他们的使用感受是评定一个系统好坏的根本，如果系统延迟过大，则会影响用户在使用上的体验。

2.2.2 易用性

如上述所说，由于系统主要面对的使用人群是用户，而他们大多是普通人，考虑到他们可能不太熟悉互联网或者计算机。因此，系统的操作方面必须遵循简单易用的原则和具备友好的用户界面，并具有相对人性化的人机交互页面，保证系统的易用性。

2.2.3 安全性

如今互联网的世界已经是非常危险的，如果想要获取他人的信息也是非常容易的，因此，系统的安全性此时就显得尤其重要。必须保证系统的安全性合格，才用使得用户放心的使用并填写自己的信息。系统提供了基于 MD5 的加密功能，每一个注册的用户密码是经过加密的，有效防止账号盗窃的可能性，保证用户信息的安全性。

2.3 可行性分析

2.3.1 技术可行性

本系统基于 Jsp+SSM+Tomcat+MySQL， 并采用 B/S 模式，由于 JSP 和 SSM 具有强大的功能，并且 Tomcat 和 MySQL 灵活且易于维护，因此它们开发方便，快速且灵活，并在实践中得到广泛使用，因此使用 JSP，SSM，Tomcat 和 MySQL 是

开发的最佳组合 轻型平台，说明该系统在技术上是可行的。

在硬件方面，随着科学技术的飞速发展，硬件的更新换代也是越来越快，从以前的机械硬盘发展到了固态硬盘，从 1G 内存发展到了 32G 内存甚至更多，已经远远超出了我们日常需求的范围，而且安全可靠，价格也更加实惠。因此硬件平台完全可以满足我们二手闲置物品系统的开发要求。

2.3.2 操作可行性

本系统是基于 B/S 结构的二手闲置商品交易网站，我们在设计的时候充分考虑到了各个年龄段用户的使用习惯，定制了一个通俗易懂的界面风格样式，让用户可以一目了然，找到他们所需要的功能，即使对互联网不了解的人，经过简单的教学，也会马上明白。至于后台界面，我们会有专门的说明方案以及文档，教会管理员如何操作后台。所以，本系统在操作方面是可行的。

2.3.3 设计可行性

本项目的总体设计采用 MVC 框架模式，能够很好的将界面显示逻辑、业务处理逻辑 和数据访问逻辑有效分离，实现各层之间的松耦合，提高相互调用的灵活性，这样的设计对系统的总体层次也划分的一清二楚，所以在设计上是可行的。

2.3.4 经济可行性

经济可行性主要的分析网站的开发成本以及维护成本，网站的作用是否可以帮助人们减轻负担。

（1）网站所需费用。由于本网站系统是立足于普通人群，开发成员都是我自已一个人完成，设备使用自己的电脑，所以不需要很大的开发成本。

（2）调查发现，网购的人群大部分都是年轻人，而大部分的年轻人都会有冲动消费的习惯，导致许多对他人有价值的商品在他们手中没有用处。也有一些年轻人喜欢购买性价比商品，二手商品则成为了他们不二之选。因此，开发一个二手闲置商品交易网站是非常有必要的。不仅方便了买家和卖家之间的需求，还可以让商品发挥出自己的价值，促进经济发展，提高生活品质。

第三章 系统设计

3.1 系统架构

二手闲置物品交易网站采用 Spring MVC+Spring+Mybatis 框架开发,是目前开发中最常用的 MVC 模式。在如今的互联网规划下, MVC 模式也从原来的三层划分发展成为了四层结构,目前通常使用的四层结构分别是: View 层, Controller 层, Service 层, DAO 层四层。从开发结构上来说整个系统架构运行流程如图 4-1 所示:



图 4-1 系统架构运行图

View 层: 就是我们所写的前端界面,负责展示内容以及与用户交互,与 Controller 相配合,形成完整的前后端。

Controller 层: 相当于一个前端控制器,由 Springboot 提供支持,主要负责映射我们的地址,形成完整的逻辑视图名,以及进行一些数据的处理。

Service 层: 逻辑业务层,为 Controller 调用 DAO 层提供接口,主要是实现一些业务逻辑的操作,这样我们就可以减轻 Controller 层的负担,使它可以更专注于处理前端页面的调用。

Dao 层: 数据持久化层,可以直接操作数据库,主要就是实现 Controller 层中需要对数据进行增删改查的方法。

3.2 系统功能模块设计

实质上,二手闲置物品交易网站的综合性相对较强,复杂程度相对较高,可对现有软件进行充分利用,进行系统设计与规划。构建完善成熟的二手闲置物品交易网站,其中涉及到以下内容,即前台网页界面、处理程序、MySQL 后台数据库系统等,在网站页面中显示出以下内容,例如商品图片、商品名称、商品价格、

商品信息等。处理程序其实也就是对用户提交表单与相关操作进行处理，存储在后台数据库的信息有用户数据、商品数据、订单数据和钱包数据等。

因此，二手闲置物品交易网站需要设置前台以及后台，其中，前台可以实现的功能有：用户注册、用户登录、钱包管理、商品推荐、商品分类、商品搜索、订单管理、商品发布、商品详情、商品求购、和个人设置。

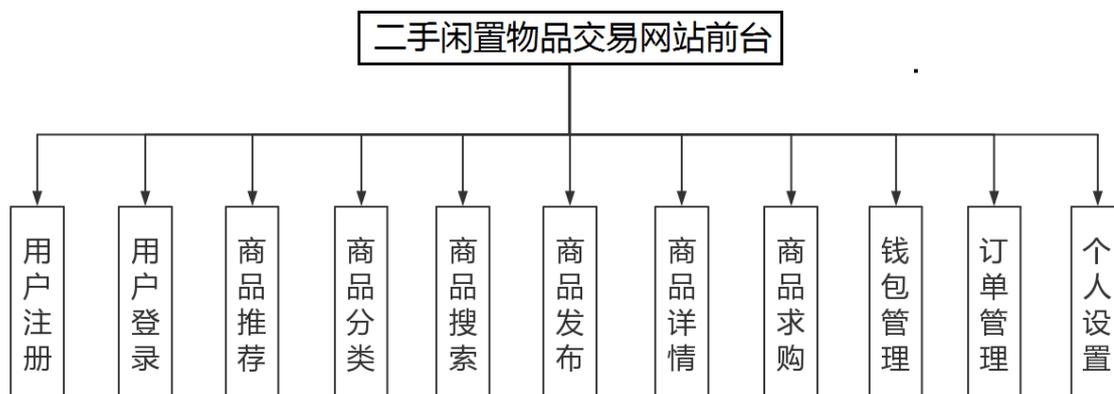


图 4-2 网站前台功能模块结构图

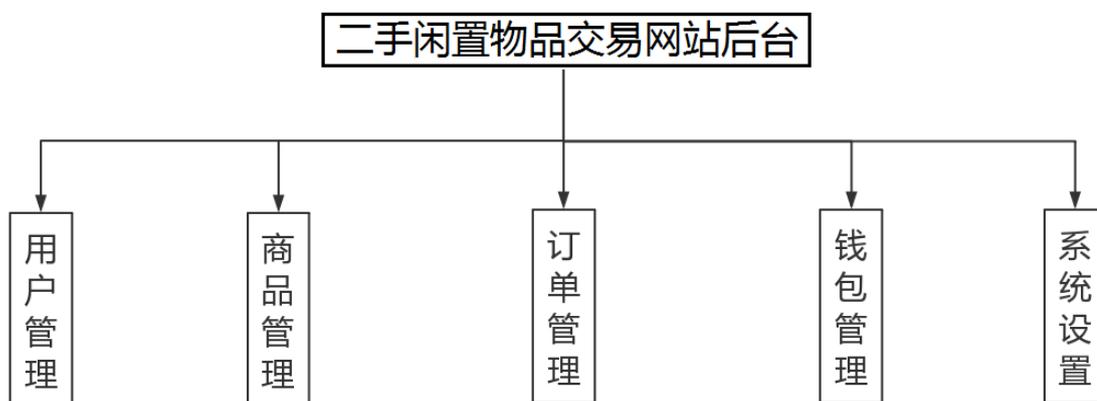


图 4-3 网站后台功能模块结构图

3.3 系统工作流程设计

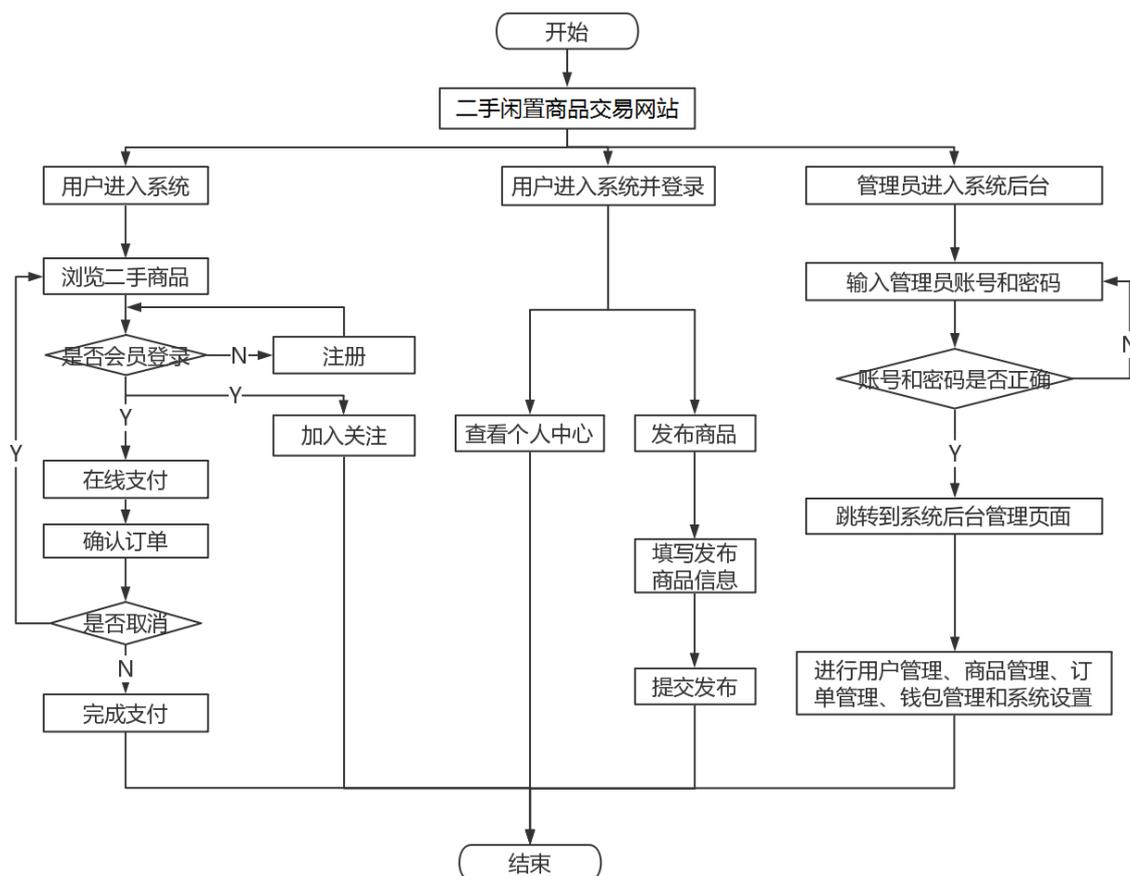


图 4-4 工作流程图

3.4 数据库设计

3.4.1 概念结构设计

概念化模型是一种建立信息世界模型的概念，它能方便、准确地表达信息世界的一般概念。概念性模型有许多表示方法，P. P. Chen 在 1976 年提出的方法是人们最广泛应用的，即实体——联系法。通过实体——联系法，我们可以使用 E-R 图完成我们对概念数据的需求模型。根据概念数据模型分析流程，最终绘制二手闲置物品交易网站的全局 E-R 模型如图 3-1 所示。

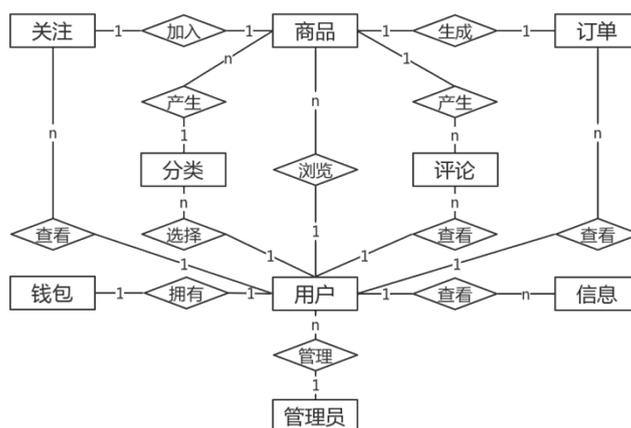


图 3-1 系统全局 E-R 图

3.4.2 逻辑设计结构

本系统数据库名称为 db_secondhandmarket，数据库中包含：1) 用户表 (user) 2) 管理员表 (admin) 3) 商品表 (goods) 4) 关注表 (focus) 5) 订单表 (orders) 6) 分类表 (catelog) 7) 评论表 (comments) 8) 消息表 (notice) 9) 钱包表 (purse)。各表数据结构如下：

(1) 用户表(user)，存储用户信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	用户 ID
phone	char(11)	N	Y	手机号
username	varchar(30)	N	Y	用户名
password	char(32)	N	Y	密码
QQ	varchar(12)	N	Y	QQ 号码
create_at	varchar(20)	N	Y	创建时间
goods_num	int(11)	N	N	发布过的物品数量
power	int(10)	N	N	信用分，普通用户默认为 100
last_login	varchar(20)	N	Y	最近一次登陆时间
status	tinyint(4)	N	N	账号是否冻结，默认 0 未冻结

(2) 管理员表 (admin), 存储管理员信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	管理员 ID
username	varchar(25)	N	Y	账号
password	varchar(25)	N	N	密码
phone	bigint(25)	N	N	手机号
userRole	varchar(25)	N	Y	角色

(3) 商品表(goods), 存储商品信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	商品 ID
catelog_id	int(11)	N	Y	商品分类 ID, 外键
user_id	int(11)	N	Y	用户 ID, 外键
name	varchar(50)	N	Y	商品名称
price	float(11,2)	N	Y	出售价格
real_price	float(11,2)	N	Y	真实价格
start_time	varchar(25)	N	Y	发布时间
polish_time	varchar(30)	N	Y	擦亮时间, 按该时间进行查询, 精确到时分秒
end_time	varchar(25)	N	Y	下架时间
describe	text	N	Y	详细信息
status	int(11)	N	N	状态 上架 1 下架 0

(4) 关注表(focus), 存储关注商品信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	关注 ID
goods_id	int(11)	N	Y	商品 ID, 外键
user_id	int(11)	N	Y	用户 ID, 外键

(5) 订单表 (orders), 存储订单信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	订单 ID
user_id	int(11)	N	N	用户 ID
goods_id	int(11)	N	N	商品 ID

order_num	bigint(25)	N	Y	订单编号
order_price	int(11)	N	Y	订单价格
order_state	int(11)	N	N	订单状态 1 待发货 2 待收货 3 已完成
order_informa tion	varchar(50)	N	Y	订单信息
order_date	varchar(50)	N	Y	下单时间

(6) 分类表(catelog)，存储分类信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	分类 ID
name	varchar(30)	N	Y	分类名
number	int(11)	N	N	该分类下的商品数量
status	tinyint(10)	N	N	分类状态, 0 正常, 1 暂用

(7) 评论表 (comments)，存储评论信息

字段名称	数据类型	主键	是否空	说明
id	int(11)	Y	N	评论 ID
user_id	int(11)	N	Y	用户 ID, 外键
goods_id	int(11)	N	Y	商品 ID, 外键
content	varchar(255)	N	Y	评论内容
create_at	varchar(250)	N	Y	评论时间

(8) 消息表(notice)，存储消息提醒信息

字段名	数据类型	主键	是否空	说明
id	int(11)	Y	N	消息 ID
user_id	int(11)	N	Y	用户 ID, 外键
context	text	N	Y	信息内容
create_at	varchar(25)	N	Y	推送信息时间
status	tinyint(4)	N	Y	状态, 0 未读, 1 已读

(9) 钱包表(purse)，存储钱包信息

字段名	数据类型	主键	是否空	说明
id	int(11)	Y	N	钱包 ID

user_id	int(11)	N	N	用户 ID, 外键
balance	float(11,0)	N	N	总钱数
recharge	float(11,0)	N	Y	充值钱数
withdrawals	float(11,0)	N	Y	提现钱数
state	int(11)	N	Y	状态 0 未审核 已审核 (1 不通过 2 通过)

3.5 关键代码

本系统使用 springMVC，sping-mvc.xml 是配置 springMVC 的，而 springMVC 是整个系统最重要的一部分。它主要是用来配置 action 的，也就是请求处理之类的。

sping-mvc.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

<!-- 自动扫描该包，使 SpringMVC 认为包下用了@Controller 注解的类是控制器 -->
<context:component-scan base-package="com.ldu.*" />

<!-- 处理请求时返回 json 字符串的中文乱码问题 -->
<mvc:annotation-driven>
  <mvc:message-converters>
    <bean class="org.springframework.http.converter.StringHttpMessageConverter">
      <property name="supportedMediaTypes">
        <list>
```

```

        <value>application/json;charset=UTF-8</value>
    </list>
</property>
</bean>
<bean id="fastJsonHttpMessageConverter"
class="com.alibaba.fastjson.support.spring.FastJsonHttpMessageConverter">
    <property name="supportedMediaTypes">
        <list>
            <value>application/json;charset=UTF-8</value>
        </list>
    </property>
</bean>
</mvc:message-converters>
</mvc:annotation-driven>
<!-- 静态资源访问（不拦截此目录下的东西的访问） -->
<mvc:resources location="/js/" mapping="/js/**" />
<mvc:resources location="/css/" mapping="/css/**" />
<mvc:resources location="/img/" mapping="/img/**" />
<mvc:resources location="/fonts/" mapping="/fonts/**" />
<mvc:resources location="/font-awesome/" mapping="/font-awesome/**" />
    <mvc:resources location="/pages/" mapping="/pages/register.jsp" />
<mvc:resources location="/WEB-INF/pages/error/" mapping="/WEB-INF/pages/error/**"/>
<mvc:resources location="/upload/" mapping="/upload/**"/>

<!-- 视图解析器 -->
<bean id="viewResolver"
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/WEB-INF/pages/"></property>
    <property name="suffix" value=".jsp"></property>
</bean>
<!-- 文件上传 -->
<bean id="multipartResolver"
    class="org.springframework.web.multipart.commons.CommonsMultipartResolver">
    <!-- 设置上传文件的最大尺寸为5MB -->

```

```
<property name="maxUploadSize">
    <value>5242880</value>
</property>
</bean>
</beans>
```

数据库属性文件：jdbc.properties

```
druid.driverClassName=com.mysql.jdbc.Driver
druid.name=squirrel
druid.url=jdbc:mysql://localhost:3306/db_secondhandmarket?useUnicode=true&characterEncoding=UTF-8
druid.username=root
druid.password=
druid.filters=stat
druid.maxActive=20
druid.initialSize=1
druid.maxWait=60000
druid.minIdle=10
druid.maxIdle=15
druid.timeBetweenEvictionRunsMillis=60000
druid.minEvictableIdleTimeMillis=300000
druid.validationQuery=SELECT 'x'
druid.testWhileIdle=true
druid.testOnBorrow=false
druid.testOnReturn=false
druid.maxOpenPreparedStatements=20
druid.removeAbandoned=true
druid.removeAbandonedTimeout=1800
druid.logAbandoned=true
```

注册页面前端 jsp 代码：register.jsp

```
<%@ page language="java" import="java.util.*" pageEncoding="UTF-8"%>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
```

```
<%
    String path = request.getContextPath();
    String                basePath
request.getScheme()+"/"+request.getServerName()+":"+request.getServerPort()+path+"/";
%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
<title>注册的面面</title>
<script type="text/javascript" src="<%=basePath%>js/jquery-3.1.1.min.js"></script>
<script type="text/javascript">
    $(document).ready(function(){
        //异步验证
        $("#account").blur(function(){
            var phone=$(this).val();
            $.ajax({
                url:'<%=basePath%>user/register',
                type:'POST',
                data:{phone:phone},
                dataType:'json',
                success:function(json){
                    if(json.flag){
                        $("#errorAccount").html("账号已被注册，请重新输入!");
                        $("#register").attr("disabled",true);
                    }else{
                        $("#errorAccount").empty();
                        $("#register").attr("disabled",false);
                    }
                },
                error:function(){
                    alert('请求超时或系统出错!');
                }
            });
        });
    });
};
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/196052032043010110>