

本科毕业设计说明书（论文）

**（2016届）**



论文题目 居家康复服务平台的设计与实现

作者姓名 陈继赟

指导教师 王春平

学科(专业) 软件工程1305班

所在学院 计算机科学与技术学院

提交日期 2017年6月

摘要

电子商务在二十一世纪以来发展迅速，各种电子商务平台层出不穷。电子商务在零售交易中的分量越来越重，逐渐取得了很大一部分的份额，二手交易作为一种从古至今存在的且具有重要意义的交易方式，在这个互联网高速发展的时代，如何让二手交易这种模式与电子商务的潮流有效结合是一个值得研究的问题。

本次毕业设计的目标是开发一个二手交易平台，基本功能包括客户登录注册、在线聊天、项目发布、交易以及交易管理等。平台实现和开发采用了当前流行的移动端混合开发模式，运用了如ionic，nodejs等新兴技术；数据库采用了mysql数据库。

虽然在市场上已经出现了相关方面的产品和平台，但是相对来说，二手交易在电子商务的发展和应用还是比较薄弱的。在中国这个资源消费大国，如何处理二手和陈旧物品将是一个很大的潜在市场。在这样的大环境下，基于移动终端的二手交易平台的开发和应用，有一定的现实意义。

**关键词：**二手交易，混合开发，node.js，ionic，Angular.js

Abstract

Electronic Commerce is rapidly developed in twenty-first Century, various e-commerce platforms are developed. Electronic Commerce is playing more and more important parts in modern market. As a pattern of Electronic commerce, Second-hand trading shows its own values from ancient time to now. It’s necessary to study how to use Second-hand trading on Internet, especially in the era of Internet.

The objective of design tasks is to study and develop an application platform for second-hand trading, the essential functions includes l register and login for customers, online communication, products demonstrations, transaction and transaction management, etc. some typical technologies, such as Node.js and ionic, are adopted in implementation of the application, and MySQL is chosen as the database.

Application of Second-hand trading on Internet is somehow lagging behind comparing with others booming applications on Internet. According to the resources consumption situation, the Second-hand trading should be a big potential market in our country. The developing and application of Second-hand trading platform based on mobile terminal is practical and meaningful.

**Keywords：**Second-hand trading, Hybrid App, node.js, ionic, Angular.js

目录

摘要 I

Abstract I

图目录 iii

表目录 iv

第一章 绪论 1

1.1 引言 1

1.2 研究背景 1

1.2.1 电子商务发展 1

1.2.2 二手交易市场的发展 3

1.3 国内外研究现状及研究意义 4

1.3.1 国内外发展现状 4

1.3.2 研究意义 5

1.4 本文的组织结构 5

1.5 本章小结 6

第二章 方法和技术 7

2.1 相关技术选型及介绍 7

2.1.1 混合开发框架Ionic以及PhoneGap平台 7

2.1.2 后台服务器语言Nodejs 10

2.1.3 关于AngularJs框架 11

2.2 项目整体设计 12

2.3 本章小结 12

第三章 系统需求和总体设计 13

3.1 系统功能需求与设计 13

3.2 数据库E-R图 14

3.3 系统安全性 15

3.4 本章小结 16

第四章 详细设计 17

4.1 登录注册功能 17

4.2 商品发布功能 18

4.3 用户在线聊天功能 19

4.4 充值购买流程 21

4.5 数据库表设计 22

4.6 本章总结 24

第五章 系统实现与测试 25

5.1 注册登录 25

5.2 聊天模块 29

5.3 商品发布 31

5.4 商品购买充值 33

5.5 本章小结 34

第六章 总结与展望 35

6.1 总结 35

6.2 展望 35

参考文献 37

致谢 39

# 图目录

[图 1‑1 货币演变历史 2](#_Toc453357337)

[图 2‑1 nodejs的异步io模型 11](#_Toc453357338)

[图 3‑1 系统功能模块图 14](#_Toc453357339)

[图 3‑2 系统数据库E-R图 14](#_Toc453357340)

[图 3‑3 MD5加密代码 15](#_Toc453357341)

[图 4‑1 注册登录功能流程图 17](#_Toc453357342)

[图 4‑2 商品发布流程图 19](#_Toc453357343)

[图 4‑3 聊天逻辑流程图 20](#_Toc453357344)

[图 4‑4 支付充值流程 22](#_Toc453357345)

[图 5‑1 登录成功和失败界面 26](#_Toc453357346)

[图 5‑2 注册成功和失败界面 27](#_Toc453357347)

[图 5‑3 注册功能核心代码 28](#_Toc453357348)

[图 5‑4 登录功能核心代码 28](#_Toc453357349)

[图 5‑5 聊天功能测试界面 29](#_Toc453357350)

[图 5‑6 聊天功能socket服务核心代码 30](#_Toc453357351)

[图 5‑7 发布功能测试 31](#_Toc453357352)

[图 5‑8 上传发布功能核心代码 32](#_Toc453357353)

[图 5‑9 购买功能测试 33](#_Toc453357354)

[图 5‑10 购买功能核心代码 34](#_Toc453357355)

# 表目录

[表 1 移动开发三种模式对比 9](#_Toc453356774)

[表 2 users表字段功能 23](#_Toc453356775)

[表 3 account表字段功能 23](#_Toc453356776)

[表 4 bill表字段功能 23](#_Toc453356777)

[表 5 wares表字段功能 24](#_Toc453356778)

# 第一章 绪论

## **引言**

电子商务是互联网时代产生的一种新型交易模式，它的出现，是为了解决交易过程中时间，距离等客观问题。在没有互联网的时代，人们之间的交易活动，都是线下实际进行的，在任何地方，任何国家，线下交易是一种习以为常的，容易接受的方式。

但是，线下交易这种传统的交易模式，也有着它本身不可避免的缺陷。比如需要的商品在周边很难采购，只有在特定的地方才有售卖。这样就会导致购买极度的不方便。又或者同一种类的商品太少，导致购买者的选择非常有限，只能将就购买。又或者某些商品平时用到不多，需要购买的人一时很难找到能购买该商品的商铺…等等这些问题，都催生了电子商务的发展，这是对人们需求的一种满足。

而在电子商务中，有一种特别的交易——二手交易。人们对二手交易的需求其实非常巨大，因为二手商品廉价、实用，很容易满足人们对物美价廉的追求。另一方面，二手交易也可以满足人们对陈旧物品的处理，既避免了浪费，又多了一笔收入。但是可惜的是，这块市场并没有引起各大电子商务平台的重视。

本文在研究，参考了许多类似电商平台后，在前人的研究基础上，设计研究出了一个二手交易的平台，给人们提供了一种购买和处理二手物品的渠道。

## **研究背景**



### 电子商务发展

在现在这个社会中，在线交易、电子商务已经渐渐充满了我们的生活。所谓电子商务[1]，较为官方的定义是以互联网信息技术为基础，依靠互联网这个平台进行商品交换的商务活动。它是一种以网上交易为支付手段的一种交易活动，是对传统意义上的商业活动的一种互联网化，电子信息化。

电子商务这个概念，我们可以从两方面来理解。所谓电子，指的就是不局限于传统的货币支付，线下交易，而更多地利用科技发展所带来的信用卡，柜员机等新兴支付方式来实现交易。而发展至今，更是出现了如支付宝，微信支付，银联支付等等更为快捷方便的支付方式。这些支付方式普遍为大众所接受，究其原因，无非是它提供了传统支付方式不能提供的快捷与便利。

而商务这个概念，指的就是从古至今就存在的商业活动。在人类社会中，商业活动是必不可少的，从远古时代的以物易物，到后来贝壳有了货币功能，渐渐有了金属货币和纸质货币，直至今日的虚拟货币。人类的商业交易活动从来就没有停止过（见图 1-1）。



图 1‑1 货币演变历史

电子商务是将“电子”和“商务”两者有机结合的一种模式，是时代科技发展的必然趋势。从电子商务概念刚刚出现的二十世纪八九十年代开始，电子数据的交换，成为了电子商务的起步期。八十年代，随着信用卡、自动柜员机和电话银行的逐渐被接受和应用，这些也成为电子贸易的组成部分[2]。进入二十一世纪以后，随着web 方面技术的迅猛发展，以及https安全协议的出现，电子商务更是越来越完善。

亚马逊，eBay，阿里巴巴等世界知名的电子商务公司通过电子商务这个模式，彻底改变了现代人生活的方式。坐在家里动动手指就能够完成一笔交易，这在以前是完全不可想象的。而现在却成为了再正常不过的事。据中国电子商务研究中心(100EC.CN)监测数据显示，2015年第三季度，淘宝的GMV为4380亿元(约合690亿美元)，同比增长15%。由此可见，电子商务已经发展到了一个很高的水平，逐渐抢占了传统交易市场的很大一部分份额。

### 二手交易市场的发展

二手交易这一特殊的交易模式，在我国是自古以来都是存在的。早在宋、元年间就有了旧货行，以及旧货贸易集市的雏形[3]。而在历史发展过程中，也出现了像典当行这样专门从事二手物品交易的场所。随着时代的发展，二手交易模式一直存在于市场中。直到今日，仍然能找到许多二手物品的聚集交易场所，很多人都喜欢去二手市场淘货。

分析其原因，有以下几点：

1、符合勤俭节约的传统美德。中国是一个崇尚勤俭节约的国家，铺张浪费一直是社会所反对的。在日常生活中，经常会有一些陈旧的物品，或者是暂时用不到的物品。最好的处理方法不是丢弃，而是利用它创造二次价值。比如说拆卸过后利用零件，或者是发掘它另有的功能，而最常见也是最好的处理方式，是二手交易。

2、符合可持续发展的方针。可持续发展就是充分利用资源，满足当下时代人们对资源需求的同时，也不会破坏伤害到未来人们对资源的需求。充分发展二手交易，可以让物品二次流入市场，产生更多的价值和作用，与此同时也避免了对新资源的过度索取和需求，非常符合可持续发展的方针。对于中国以及整个人类社会来讲，这都是值得推广的。

3、满足贫困人群对资源的需求。据国家统计局数据显示，截止2015年，我国贫困人口数量仍然有7017万之多，占到农村总人口的7.1%。可见我国的贫困现状仍然不容乐观，而对于这些消费能力不高，生活水平低下的贫困人口，二手物品无疑是他们最好的选择。这让人们能以尽量低廉的价格买到自己所需的物品，满足基本的生活需求。

## **国内外研究现状及研究意义**



### 国内外发展现状

进入二十一世纪以来，在各国政府人民的推动下，互联网发展迅速。相应的，电子商务也随之逐渐成长，取得了不小的成就。在美国等经济发达的欧美成熟国家中，已经有了非常完善的电子商务、网上购物网站。比如说著名的亚马逊，eBay等世界知名公司，他们的业务已经遍布全球，让整个世界成为了一个市场。

仔细分析亚马逊等公司的发展历程，我们可以看见互联网的迅猛发展对电子商务的促进是明显的。互联网是电子商务存在的依赖，想要发展好电子商务，互联网是基础。互联网把全球各地的人们联系在了一起，让全球性的贸易成为现实。而相应的，单独面向二手物品的电子商务网站也层出不穷，如eBay，craigslist.org等等。通过这些平台，人们可以轻松地从全世界的范围内获取自己想要的二手物品，或者是出售自己用不着的旧东西。可以说，在国外一些发达国家，这方面的发展已经非常完善了。

而在国内，我们同样拥有像阿里巴巴，京东等成熟的电商平台，也有像蘑菇街，唯品会这样发展中朝气蓬勃的年轻企业。随着今年来国家政策对互联网以及电子商务方面的支持，国内这方面的发展也是有目共睹。阿里巴巴作为民营高科技企业，把电子商务的生意做到了全球，让全世界都知道了这家公司。这些种种，也显示了国内良好的发展氛围。

因为这些电商企业，电子商务公司的迅速崛起，国内消费模式也有了巨大的改变。网络购物成为了新潮的购物方式，特别吸引了许多年轻人去尝试。在网络购物环境下，消费者可以集中时间搜集、查找大量有关商品的信息[4]。这样一来，便利的购物体验一下子就改变了传统的购物思维。

但是对于二手交易这一块市场，国内许多电子商务平台并不是很重视。要么是专注于二手车，二手房这样更为细致的划分，要么就是在原有平台上添加一部分二手交易的功能。真正去做二手交易这一块市场的电子商务平台屈指可数，只有如闲鱼，转转等少数几款平台。并且这几款产品的知名度和宣传力度并不大，导致真正有二手交易需求的人无法及时找到以及利用该平台。国内电子商务平台对二手交易市场的不重视，导致了二手交易平台发展缓慢，成熟度不高，使用率也迟迟上不去。

如何让网上二手交易市场真真正正的发展成熟起来，是当下值得研究探讨的问题。互联网技术发展至今，其范围已经逐渐扩大，再也不仅仅限于电脑PC端了。智能手机以及各种平板电脑的出现，让互联网的触手伸向了移动终端。现如今，随着安卓、IOS平台的崛起，移动互联网成了新宠儿，这个新兴的时代潮流正是二手交易在国内网络化的一个重要契机。借助于移动互联网的发展，让二手交易从普通电子商务中独立出来，成为一块单独的市场，这是非常有发展前景的。

### 研究意义

本课题在于研究二手交易这种特殊的交易模式在电子商务环境下的实现和发展。通过对市场上一些已有平台的研究分析，采取合适的开发技术和架构，构建模拟出一款在线交易二手物品的移动终端软件。

本课题的研究开发有利于发展、重视二手交易的价值所在，对该特殊交易模式的网络化有着推动作用。对市场该类软件的开发研究也有一定借鉴作用。

## **本文的组织结构**

本文一共分为五章，研究讨论了二手交易在电子商务时代的困境和发展，以及国内外对该方面的研究现状，并根据各种材料的研究分析，设计开发出一款模拟二手交易的软件，各章内容如下：

第一章，详细介绍了本次课题的背景，相应国内外对此领域的研究开发，以及本次课题的内容和意义。

第二章，详细介绍了本次研究开发所需要用的技术架构以及总体概括设计。为之后的详细模块设计做基础。

第三章，重点介绍了各个模块功能的详细设计，技术难点，以及部分代码实现情况。

第四章，主要介绍了各个功能的测试情况，以及最后软件完成的界面情况，并作出分析和说明。

第五章，对本次研究开发做出一个总结，并分析发展情况和存在问题。

## **本章小结**

本章主要介绍课题的研究背景、在国内外相关领域发展情况，现状以及项目的主要内容和研究意义。最后，说明了本文的组织结构。

# 第二章 方法和技术

要设计开发出课题所需要的项目，首先就是要确定技术选型。经过对市面上一些相关产品平台的研究和了解，我决定使用如今比较新潮的移动端混合开发框架作为整个项目的整体架构。继而选用发展蓬勃，潜力很大的node.js作为后台服务器开发使用的语言。数据库选用了成熟的mysql数据库。在真正开始开发和研究之前，我们必须要对这些技术有一个整体的了解。以下就来详细介绍一下本课题开发研究过程中所使用到的一些技术，以及本项目粗略的一个整体设计。

## **相关技术选型及介绍**



### 混合开发框架Ionic以及PhoneGap平台

首先先介绍一下移动端开发的三种模式。

1、原生开发

Native APP[5]，又称为原生开发模式，这种开发模式和系统PC桌面程序相似，它在平台运行时会直接调用所在平台硬件API。所以在进行原生开发时，往往会要求开发人员对硬件设备和所用平台的性能结构等比较熟悉。开发人员必须针对不同型号的不同硬件设施进行了解。原生开发采用较为底层的软件语言，这种开发模式和嵌入式系统的开发比较相似。比如运行在ios系统上的苹果应用，开发人员需要用到Objective-C或者swift等编程语言，而对谷歌的安卓系统开发则需要用到JAVA这门语言。

这种原生模式结构有两部分组成：服务器以及APP客户端，APP应用使用的UI元素、数据、逻辑框架等都安装在移动终端上。原生开发模式的优点是应用程序位于平台层上方，兼容能力比较好，能够方便的调用硬件设施：如摄像头，相册，拨号等等，充分利用设备本身的资源，有着很好的交互体检。但是原生开发模式的缺点也比较明显，开发的周期很长，对于开发人员的知识和技术水平要求很高，开发成本大。而且往往只能针对不同的系统进行二次开发，平台移植性较差。

2、web开发模式

Web App[6]，使用的是Html,Js,Css等WEB前端的编程技术，程序运行是基于浏览器运行的，通过浏览器来调用移动端平台的API。APP不需要安装，对不同硬件设备的适应能力很强，这一点上比原生开发模式要优秀。Web开发模式具有跨平台的优势， APP 移动客户端只是安装了程序的框架部分，而程序运行所需要的速度是从云端实时取出的。 Web App 模式有以下优点：平台移植性很好，开发周期很短，开发所需要的成本不高，当你的应用需要更新时，我们只需要将服务器端代码更新，而不用同步将客户端的代码更新。这样一来，就大大减少了工作量，而且软件的更新速度也很快。但是由于是基于浏览器的，所以Web App始终做不到像原生应用一样流畅，以及良好的交互性体验。并且在调用移动端本地文件系统和硬件接口的性能也比较弱，并且当用户使用时需要消耗流量来进行一些数据交互，如果没有网络，会导致数据交互无法进行，而且使用流量也限制了用户的使用环境。

3、混合开发模式

Hybrid App 模式是介于 Web App、Native App 这两者之间的一种开发模式，又被称为混合开发模式。它表面上看来非常像一个原生APP，但是实际上它只是套了一个 UI Web View，在里面访问的是一个 Web App。混合开发模式和web开发模式一样，入门门槛不高、占用系统资源比较少；而且用户体验明显优于 Web APP，而且和原生开发模式相比，整个应用也显得轻巧，但相反的，混合模式却能够实现媲美原生开发的丰富功能和优良的交互体验，并且有良好的平台移植性，这是原生和web的有效结合。相比于 webapp，hybrid app 把 Web代码封装到 App 里面可以提高性能和可访问性，但是更新仍然要重新下载更新。混合开发模式，在市场上主要有Phone Gap，AppCan，appMobi，Titanium 等平台 ，它们是基于webkit 开源内核，基于H5 标准开发，对不同平台有良好的支持，支持调用开发者自定义的插件，成为移动开发者的首选开发平台。

上述三种的移动开发模式，原生开发和web模式开发都有明显的缺陷，所以我们采用混合开发模式来完成整个系统。

以下是三种开发模式的对比，如表1：

表 1 移动开发三种模式对比

|  |  |  |  |
| --- | --- | --- | --- |
|  | 原生开发 | 混合开发 | Web开发 |
| 优势 | 流畅，体验好，原生API调用方便 | 开发流程短，能很好模拟原生调用 | 开发简单，易于开发 |
| 缺点 | 开发流程长，技术要求高 | 调用原生底层不如原生开发流畅 | 对网络要求高，不能很好模拟原生 |

Ionic框架和Phone Gap[7]平台：Ionic是目前非常火热的一款 HTML5 手机应用开发框架。通过 SASS 构建应用程序，它提供了很多有用的组件来帮助开发者轻松开发强大的应用。ionic是一个混合开发模式的框架，它的出现是为了使用WEB开发的技术，在HTML5的支持下，开发出尽量接近原生应用的项目。Ionic使用了Angular框架和Sass这个css预处理器。通过使用Phone Gap的编译平台，ionic框架开发的移动应用可以实现编译成不同平台的应用程序。在开发ionic项目时，根据项目的需求，按需添加安卓或者ios平台。在H5移动app开发中,速度是很重要的。Ionic在最新的移动设备中表现非常卓越，运行非常流畅。 尽量不操作dom，硬件加速过渡，让您感觉到用H5开发的app也可以运行的极为流畅。HTML5是由W3C发布的新一代HTML标准,不仅强化了Web网页的表现性能,还拥有很多API接口,开发者利用这些接口,可以便捷地创建更加丰富、引人入目的应用程序 。Ionic以流行的原生移动开发SDK为蓝本，使开发过原生iOS或安卓app的任何人都容易理解。开始只需书写你的代码，完成时通过Phone Gap发布。一次开发，处处运行，良好的平台适配性是其一大优势。简洁，简单，而且实用。 Ionic为所有当前移动设备而设计，并且呈现完美。伴随众多流行移动组件，结构，交互规范，以及华丽的（且可扩展）的主题，强大的组件和优秀的UI组件设计，让ionic在混合开发模式中有了自己的一席之地。在我们的系统中，ionic主要承担了移动端的界面展示，以及各种漂亮的交互设计。

### 后台服务器语言Nodejs

Node.js是一种新兴的开源技术，它将JavaScript从Web浏览器移植到常规的服务器端，使用Chrome的V8虚拟机来解释和执行JavaScript代码。Node.js 是基于[事件驱动](http://baike.baidu.com/view/536048.htm" \t "_blank)的，它提倡的非阻塞I/O模型在应对高并发需求时非常高效，它非常擅长开发数据比较密集的实时应用[8]。Node是一个用C语言开发的JavaScript运行环境(runtime)。简单来说，它对谷歌的 V8引擎进行了封装。V8引擎在执行JavaScript的速度非常快，性能十分卓越。Node优化了一些特殊的用例，代替了一些原生的API，这使得V8引擎在非浏览器环境下运行得更好。

Node.js大胆采用了事件驱动、异步编程，专为网络服务而设计。事件驱动这个概念并不是最近才兴起的，想一想我们之前接触的项目中，在进行网络编程时，我们也会使用到回调函数。Node.js的设计思想就是事件驱动，它自己的很多API都是基于事件驱动，异步编程而开发的。异步输入/输出是事件驱动编程的一个重要原因，因为它可以防止在一个输入/输出操作中的应用程序被阻塞[9]。

举个例子，比如说node.js的socket.io模块，它的connection，message，disconnect等事件都是异步的。在使用Node.js时，开发人员需要根据自己的业务逻辑注册相应的回调函数。这些回调函数都是异步执行的，所谓异步执行，就是指这些函数并不是按照他们在代码中注册声明的顺序来决定执行先后的，只有当相应的事件处理完毕触发回调时，才会被调用。

我们之所以采用事件驱动、异步编程的设计，是因为通过它，我们可以节省系统资源，不至于浪费宝贵的系统资源，执行时不必阻塞来等待前一些操作的完成，可以充分利用系统资源进行其他的操作。在后端的网络服务编程中，这个问题就更加明显了，而Node.js正是优化这个问题的一把好手。当我们进行服务器开发时，高并发的请求往往很难处理，如果采用阻塞式的函数，那很明显，我们会导致资源浪费和时间延迟。因此，在这个时候，采用node.js这种事件驱动异步编程技术就很合适了。nodejs异步io模型见图2-1。

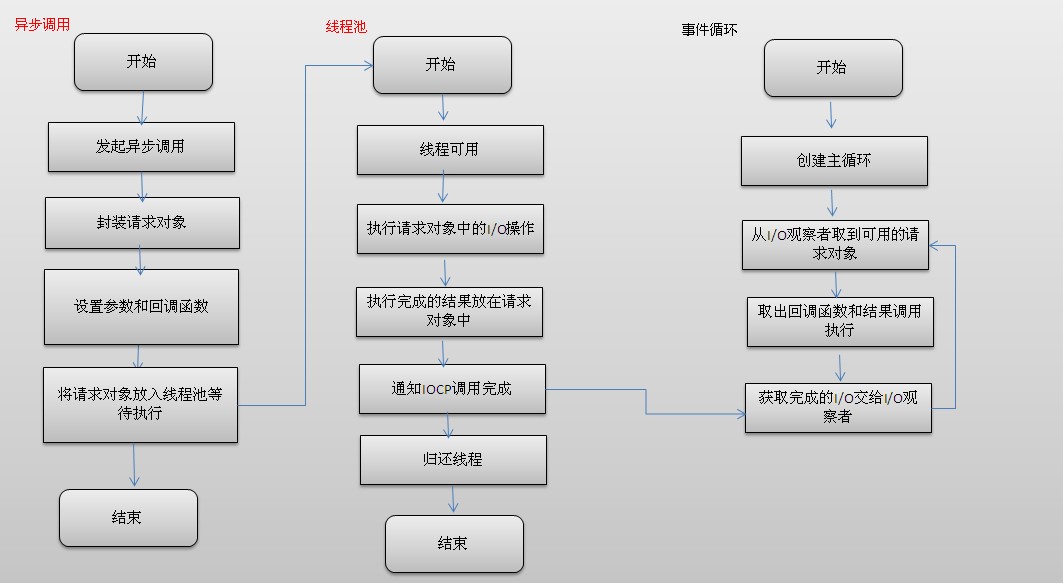


图 2‑1 nodejs的异步io模型

### 关于AngularJs框架

AngularJS[10]是Google开发者设计和开发的一套前端开发框架，帮助简化前端开发的负担。

在构建动态网页应用时，我们通常会使用一些类库，或者是框架来帮助我们更好地完成开发。类库和框架各有各的优势，在web开发的领域中，JQuery[11]无疑是一个极为优秀的前端类库，通过它，你可以简化你的操作，更快的完成开发。而框架则不一样，它是一种已经实现了的应用，在使用时，你只需要编写具体的业务逻辑代码。比较有名的框架有knockout、sproutcore，angularjs等。

AngularJS使用指令（directive）这一方法，去弥补html本身的缺陷，这种技术能让浏览器识别新的语法。AngularJS有着一些非常独特有趣的特性，比如MVVM设计模式、模块化开发、数据的双向绑定、自定义指令、依赖注入的概念等等。

在我们的系统中，angularjs主要用于客户端视图界面的业务逻辑的编写，以及与后台服务端的交互。

## **项目整体设计**

本项目整体上分为前后两大块，前端部分主要负责页面的展现，与用户的交互以及与后台服务器的数据交互。后台部分主要是负责与前端进行数据通信，并且与数据库进行数据的操作。从功能模块上来细分，主要设计完成了以下四个功能模块：用户注册登录功能，在线聊天功能，充值支付功能，商品发布功能。

## **本章小结**

本章主要介绍了开发研究过程中用到的一些主要技术，以及技术选型，如Ionic，nodejs等。在对这些技术进行简单介绍的同时，也交代了选择使用这些技术的原因和好处。另一方面，本章还简略地介绍了项目的设计思路和整体架构。为下一章的进一步详解做了铺垫。

# 第三章 系统需求和总体设计

## 系统功能需求与设计

在对已有产品的研究和分析之后，我们逐渐了解了这样一个系统需要一些什么功能。功能来源于需求，只有搞清楚用户的使用需求，我们才能有效地设计出必须的功能。

1、注册登录

在交易行为过程中，每一个购买者和出售者都是独立的个体，在这个基础上，我们需要一个功能实现用户的独立。而这样的功能就是账户的注册和登录。只有用户拥有自己独立的账户，在线购买出卖的交易行为才能够成立。

2、商品发布

商品是交易的主体，在线下交易中，商品的来源是卖家从货源处进货。那么在网络上，如何让向买家展示自己的商品就成为了卖家的第一步。如果没有商品这个主体，交易行为也就不能进行了。基于这点的考虑，我们设计了商品的发布功能。

3、在线聊天

交易行为过程中，卖家和买家一定是有必须的交流，不论是想详细了解商品或者是在价格上的交流。网络上也是不能少了这一个过程，双方的交流可以更好地让双方完成这笔生意。

4.、购买交易

设计这个系统的目的，就是模拟实现在线购买二手物品的过程。毫无疑问，最后的交易支付功能一定是必不可少的。

图3-1是本系统的功能模块图。

图 3‑1 系统功能模块图

## 数据库E-R图

在进行数据库具体表设计之前，我们必须弄清楚，在这个系统中，各个实体之间的关系，只有明白了其中的联系和方法，我们才能更好地设计数据库。

图3-2是本系统数据库的E-R图。

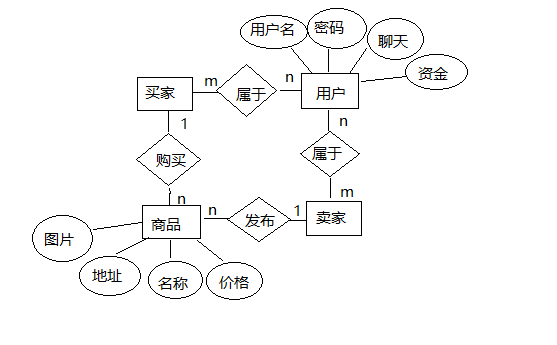


图 3‑2 系统数据库E-R图

在这个系统中，实体买家和卖家都是属于实体“用户”的，他们都拥有账户、密码、聊天对象、资金余额等属性。而对于实体商品而言，它与卖家的关系，是卖家发布商品，它和买家的关系，是买家购买商品。商品这个实体拥有商品名称、商品价格、商品地址、商品图片等四个属性。

## 系统安全性

在本系统中，设计并不是很复杂，需要考虑到的安全性方面的问题，最直接明显的就是用户的密码安全问题。用户的密码被存在后台的数据库中，一旦数据库泄露，大量的用户账户密码数据将被不法分子获得，这是极其危险的。我们该如何解决这个问题呢？

在以前的数据库设计中，通常会把用户密码这样重要的数据直接明文存储，这样的安全系数非常低。我们决定在用户注册时，首先将用户的密码加密过后再存储到数据库中。这样一来，即使数据库的数据泄露，不法分子也无法使用用户的账号登录。

而对于加密算法，本系统采用了广为使用的MD5加密，这种加密方法是不可逆的，也就是说想从密文反解出明文几乎不可能，而暴力字典破解也需要花极长的时间，几乎可以说是无法暴力破解。图3-3是应用MD5加密的代码。

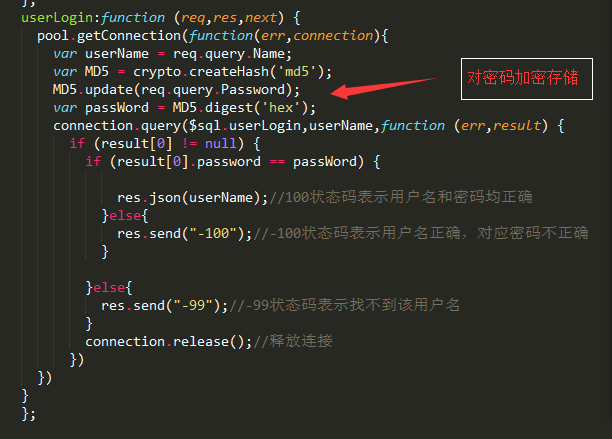


图 3‑3 MD5加密代码

## 本章小结

本章主要介绍了本系统的一些主要概要设计，包括阐述了系统设计的几个主要功能模块，对于数据库的E-R模型的分析，并给出了E-R图，最后分析了系统所要注意的安全性问题。总体来说，本章是对系统的一个简单准备和概述，为下一章对系统的主要实现进行了铺垫。

# 第四章 详细设计

经过上一章对技术知识的简单了解和介绍，以及初步的项目构建思路，我们就可以正式进入项目的设计编码阶段了。在这个阶段，我们使用既定的技术知识，实现预定的几个功能模块，并对其中遇到的问题和困惑进行了研究和讨论。接下来让我们具体分析每一个功能模块的实现。

## 登录注册功能

登录注册功能是所有功能的基础，只有注册了账号的用户才能够进行其他的操作。注册登录的具体功能流程如图4-1所示。

否 否

是 是

图 4‑1 注册登录功能流程图

在用户初次进入页面时，若用户只是游客状态，那么该用户只有浏览商品列表的功能，其余功能都不能使用。若用户想使用其他功能，系统会提示用户必须要登录才能使用其他功能，并且系统自动将界面跳转至用户登录界面。在登录界面，如果用户之前已经注册过相关账号，那么直接输入账号密码，若账号密码无误，用户则顺利登录，可以使用聊天，购买，充值，发布商品等功能。若用户还没有相关账号，那么用户可以点击注册按钮，跳转到注册页面，进行账号注册，随后再进行登录。

以上是注册登录时流程的说明。

在具体代码实现方面，用户注册账号时，点击提交按钮后，程序会获取用户输入的账户以及密码，将其发送给后台服务器。服务器获取账号密码后，根据账号信息去查询用户数据库，若该账号已经被注册，那么将会返回一条信息，告诉前台。前台会弹出弹窗提示用户该账号已经被注册，请重新注册。若在查询数据库后，发现该账号没有注册，那么后台将会用MD5加密方式将该密码加密，并且和账号作为新的一条数据插入用户表。

登录注册功能是所有功能的前提，接下来我们继续介绍分析其他几个功能模块的实现。



## 商品发布功能

二手交易平台说到底也是个网上购物平台，商品发布的功能当然是不能缺少的。在商品发布方面，首先系统一样会检测用户是否登录，未登录的用户无法使用本功能。当用户已经登录时，用户就可以根据页面提示填写需要发布商品的一些基本信息，然后点击提交发布。

在这里系统需要调用安卓移动端的原生API来实现对商品照片的选取。用户需要填写商品名称，售卖价格，商品所在地址以及商品图片等必须信息，系统会在用户提交时判断用户是否完整填写了信息。如果有信息没有填写完整，系统将会提醒用户未填写完整。若系统检测用户正确填写了信息，那么该商品信息就会发送到系统后台。后台接受到该信息后，会将商品图片保存到磁盘中，并把相关信息更新到数据库商品表中。

以下是相关流程图：

否

是

否

是

图 4‑2 商品发布流程图

在处理照片上传时，开发过程中遇到了一个问题，由于我们对于一件商品允许上传最多三张相关图片。而在上传过程中，cordova插件一次只能上传一张图片，因此我们只能循环上传。而循环上传导致的后果就是后台会将每一次上传都看作是一件新商品。这会导致商品的重复。经研究，解决方案为：在循环上传之前，给这些同件商品不同图片给定一个相同的key值，此key值为用户名字符串和当前时间字符串的拼接。后台根据图片所带的key值来判断这些图片是否为同一件商品。若是同一件商品，则数据处理后存为一条数据。

## 用户在线聊天功能

在电子交易过程中，不同于线下交易，人们可以面对面商量价格，数量等信息，很显然，这个过程是必须的。所以我们设计了一个在线聊天的功能，方便购买者和售卖者之间的交流。想要实现这个功能，我们必须要了解网络编程相关的技术，比如说socket。Socket的连接类型可以分为两种，分别是面向连接的字节流类型和面向无连接的数据报类型[12]。

在我们的项目开发中，我们使用了socket.io来实现聊天功能。Socket.io是对websocket进行了封装，并且实现了它在服务端的代码。HTML5 WebSocket的实现使Web应用不需要每次都发起HTTP请求来建立与服务端的连接，而是仅在第一次请求连接后就建立起TCP Socket连接，从而基本做到通信时的时时响应，提高了通信效率[13]。通过使用这个模块接口，我们可以完成一个在线聊天的功能。

图4-3是在线聊天的逻辑流程图。

点击用户名

图 4‑3 聊天逻辑流程图

在这个聊天逻辑流程图我们可以看到，用户想要进行聊天时，可以通过两种方式，第一种通过直接进入自己的聊天列表，与之前交流过的人继续交流，或者是直接在首页点击感兴趣的卖家的账户名字，直接进行一个新的聊天。在这个过程中，我们是如何实现点对点交流的呢？

当我们选定交流对象时，无论是哪种方式，我们会进入聊天的详细界面。A用户发出一条信息时，首先前台会将它的信息获取，并且以数组形式将聊天内容存储到本地浏览器提供的localstorage[14]。然后系统将该信息用socket发送给后台，并且在这个socket上，我们存储了该socket的所有者以及该条信息想要发送给谁。后台程序在收到这条socket信息时，首先它会判断该socket是否已经保存过，如果保存过，那就直接获取该socket所要发给的用户名，并在此socket对象找符合要求的socket。若存在，则把信息用该socket发送给对方。如果一开始的socket没有在数组中保存过，那后台程序首先会把socket保存下来，再进行之后的操作。

在实现了socket存储的同时，我们还要进行聊天列表数据的存储。当用户A发起与新用户B的聊天时，我们会把用户表中A用户的chatlist字段加上用户B的用户名，同时在B的chatlist对象上加上A的用户名。

## 充值购买流程

支付功能涉及到方方面面，如支付宝、微信等接口都需要有公司认证申请才可使用，这对本次研究来说较为困难。因此在支付方面，我们只是简单设计模拟了该功能，用户可以通过个人界面进行充值，也可使用充值后的虚拟财产进行购买行为。在整个购买流程中，后台会先判断用户账户里是否有足够金钱，如果数额不够支付该商品价格，则后台会返回信息，前台会提示用户金额不足，请充值后再购买。若用户余额足以支付该商品，那么后台将会从买家账户内扣去商品费用，并转入卖家账户内。与此同时，交易完成后，此条交易记录将会被存储下来，而该商品相关信息则从数据库中删除。

以下是支付充值的流程图。

否

是

否

是

图 4‑4 支付充值流程

## 数据库表设计

数据库设计通常包括需求分析，概念结构设计，逻辑结构设计，物理结构设计，数据库实施，数据库运行和维护六个阶段[15]。本项目实现的功能并不复杂，数据库设计也较为简单明了。一共有users,wares,bill,account四张表。users表主要用来存储用户相关的信息，wares表存储商品信息，bill表存储交易信息，account表存储用户余额信息。为了方便操作和修改数据库，在实际开发过程中我使用了Navicat for Mysql这款可视化工具。具体字段见如下几张表。

表 2 users表字段功能

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 是否主键 | 值类型 | 字段描述 |
| username | 是 | char | 用户名 |
| chatList | 否 | char | 用户聊天列表 |
| password | 否 | char | 用户密码MD5值 |

表 3 account表字段功能

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 是否主键 | 值类型 | 字段描述 |
| user | 是 | char | 用户名 |
| account | 否 | int | 用户金额余额 |

表 4 bill表字段功能

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 是否主键 | 值类型 | 字段描述 |
| buyer | 否 | char | 买家用户名 |
| ware | 否 | char | 商品名 |
| seller | 否 | char | 卖家用户名 |
| money | 否 | int | 交易金额 |

表 5 wares表字段功能

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 是否主键 | 值类型 | 字段描述 |
| price | 否 | int | 商品价格 |
| address | 否 | char | 商品地址 |
| warename | 否 | char | 商品名称 |
| owner | 否 | char | 商品拥有人 |
| imagepath | 否 | char | 商品图片存储地址 |
| flags | 是 | char | 商品唯一标识 |

## 本章总结

本章主要详细介绍分析了各个功能模块的详细设计以及使用流程图，让我们对整个系统的一个流程有了充分详细的了解。对数据库表进行了详细的设计，为下一章具体实现的解析做了铺垫。

# 第五章 系统实现与测试

在完成了大多数整体功能的设计后，我们来到了项目的实现以及测试阶段。在这个阶段，我们具体解析主要功能的实现以及测试。我们测试了各个功能模块，这种测试方法在软件测试的专业术语称为单元测试[16]。在进行完单元测试之后，我们测试各个功能模块之间的交互是否存在问题，这被称作集成测试[17]。最后我们需要对整个项目完全测试一遍，这个过程称为系统测试[18]。我们要对完成的地方加以肯定，对还不完善的地方提出疑问。



## 注册登录

首先我们先来看一下该功能完成后页面的展示是怎样的。图5-1是登录功能成功和失败的界面，图5-2是注册功能成功和失败的界面。

图 5‑1 登录成功和失败界面

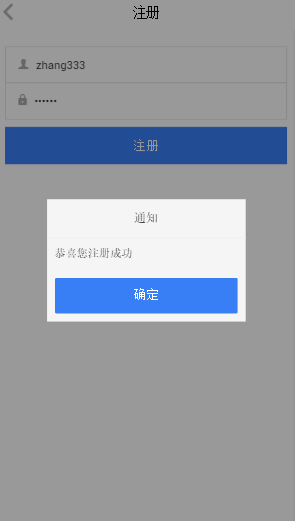
 

图 5‑2 注册成功和失败界面

经过测试，注册登录模块顺利通过测试，登录时，当用户名和密码不符合要求时，系统就会提示用户出错，而相应的注册模块也是一样，这样保证了用户名的唯一性和正确性。那我们这个功能具体是如何实现的呢？图5-3和5-4是该功能模块实现的核心代码。



图 5‑3 注册功能核心代码



图 5‑4 登录功能核心代码

从提供的核心代码我们可以看出，登录和注册功能都是从系统前台获取到用户发送请求中的参数，然后根据这些参数（密码需先加密）和数据库中存储数据进行比对，然后再根据比对结果返回给前台相应的状态和参数。

## 聊天模块

我们先来看一下聊天界面测试后是怎样的。如下图所示。



图 5‑5 聊天功能测试界面

经过对聊天功能的测试，我们发现，虽然聊天功能没有什么大问题，可以进行交流。但是仍然有值得改进的地方，比如说该聊天功能必须存储过自己的socket，如果没有，那么别人永远联系不到你。也就是说，你必须发过信息，才能让别人联系到你。这个bug虽然看起来不大，但是用户体验[19]很不好，但是由于技术所限，不能很好理解利用socket，暂时无法解决这个问题。希望以后能有更好的办法实现在线聊天功能。具体的实现核心代码5-6图。



图 5‑6 聊天功能socket服务核心代码

## 商品发布



图 5‑7 发布功能测试

如5-7图所示，是商品发布功能的界面，经过我们多次测试，发布功能各项指标都通过，没有什么问题。该功能的实现过程其实并不困难，相反还是界面上的实现过程比较棘手。我们使用了一款地址联动的插件，来帮我们实现地址的选择，然后通过cordova提供的一些插件，我们调用了安卓原生的照片库等API。在用户填写完成并点击发布后，系统会进行检测，并给出相应的处理。图5-8便是具体的实现核心代码。



图 5‑8 上传发布功能核心代码

## 商品购买充值

图 5‑9 购买功能测试

购买功能经过测试顺利通过，没有任何问题。虽然我们实现的购买充值功能只是一个模拟功能，但是我们还是很好的实现完成了。这能帮我们更好理解整个购买充值的过程。以下是具体的代码实现。



图 5‑10 购买功能核心代码

## 本章小结

本章主要对已完成的项目进行了全方位的实现和测试，经测试，在线聊天功能模块有一些瑕疵，其余模块均良好实现。在测试中，对于一些bug我也努力去修正，尽量将本项目的问题减少到最少。这是对自己，对自己课题的负责。

# 第六章 总结与展望



## 总结

本文提出了二手交易的独特性，并建议大力在电子商务方面大力发展二手交易。并且实践开发出一款基于安卓的二手交易平台。在此平台中，主要实现了电子商务一些必不可少的功能。如登录注册，购买，充值，发布商品，在线聊天等等。

在主动实践并研究过后，我更加坚定了二手交易在电子商务时代的潜力和发展前景。特别是在移动互联网迅猛发展的今天，利用不断在更新换代的移动互联网技术，将二手交易剥离出来，一定是值得发展研究的一个方向。本次研究开发虽然微不足道，但仍然提供了一些可以借鉴的地方：首先，本文着重于移动终端的开发，将移动端作为本次研究的载体，并使用了如今很火热的混合开发模式。其次，本次研究对基础的电子商务模式功能做了一个解析和实现方案的提供。这两点是本次课题的主要贡献和值得探讨的地方。

## 展望

经过本次课题的研究，我对电子商务，二手交易的发展更为看好。时代是在进步的，世界永远在发生变化。但是有一点，人与人之间的交易是永远不会停止的。交易是获取生活生存资源最为简单直接的方式，在未来的社会里，交易不会消失，电子商务也会发展得越来越快，越来越超出人的想象。而同样的，不管是移动设备的发展还是二手交易的未来，都是光明的。

追求方便快捷是人类的天性，这一点促进着移动终端的发展，几乎所有的市场咨询机构和专家都赞同，移动互联网是未来十年内最有创新活力和最具市场规模的新领域[20]。而在未来的世界中，资源必定是越来越少，越来越珍贵。在这样的社会环境下，物品的多次利用和二次交易就显得格外重要了。在不久的将来，相信科技的发展会让两者更为融洽的蓬勃发展。

# 参考文献

1. 张昊. 探索中国B2C电子商务的发展之路[J]. 科协论坛, 2007(4):121-121.
2. 彭向峰. 电子商务平台-购物网站的设计与实现[D]. 山东大学, 2012.
3. 金鹏, 黄河, 王清波. 二手货市场的发展前景分析[J]. 市场周刊:财经论坛, 2003(10):57-58.
4. 华冰. 我国网络购物发展综述[J]. 网络财富, 2010(21).
5. 王志勤. Web App还是Native App[J]. 商学院, 2012(5):16-17.
6. 陈勇. Web App现状分析及展望[J]. 通信与信息技术, 2012(4):77-78.
7. 武晶晶. 跨平台的PhoneGap研究[J]. 信息安全与技术, 2012, 3(12):71-72.
8. 朴灵. 深入浅出Node.js[M]. 人民邮电出版社, 2013.
9. Tilkov S, Vinoski S. Node.js: Using JavaScript to Build High-Performance Network Programs[J]. Internet Computing IEEE, 2010, 14(6):80-83.
10. Darwin P B, Kozlowski P. AngularJS web application development[J]. 2013.
11. Volder K D. JQuery: A Generic Code Browser with a Declarative Configuration Language[J]. Lecture Notes in Computer Science, 2006, 3819/2006:88-102.
12. 夏玲. 客户端与服务器端的Socket通信[J]. 电脑编程技巧与维护, 2009(17):49-51.
13. 李代立, 陈榕. WebSocket在Web实时通信领域的研究[J]. 电脑知识与技术, 2010, 06(10):7923-7925.
14. sessionStorage, localStorage. HTML5 Web Storage[J]. Dr Dobbs Journal.
15. 张露, 马丽. 数据库设计[J]. 安阳工学院学报, 2007(4):76-79.
16. 董威. 单元测试及测试工具的研究与应用[J]. 微型电脑应用, 2008, 24(5):24-26.
17. 夏耘, 林华. 软件系统的集成测试技术研究[J]. 计算机工程, 2000, 26(10):102-104.
18. 李志峥, 杨社堂. 基于B/S结构下的软件系统测试研究[J]. 科技情报开发与经济, 2006, 16(7):232-234.
19. 罗仕鉴, 朱上上, 应放天,等. 手机界面中基于情境的用户体验设计[J]. 计算机集成制造系统, 2010, 16(2):239-248.
20. 肖志辉. 移动互联网研究综述[J]. 电信科学, 2009, 25(10):30-36.

# 致谢

在这里我首先要感谢我的导师：王子仁老师。王老师是一位和蔼、知识渊博的老师，在毕业设计的从头到尾，都给了我很多指导和帮助，并且在生活上也十分关心照顾学生。其次我还要感谢我的同学和同事，在项目设计实现的过程中，给了我不少的帮助和建议，让我能顺利完成项目的设计。最后我也很感谢培养我四年的母校浙江工业大学，它在这四年中给了我良好的学习环境，让我能够汲取到知识和能力。