

北京交通大学

本科毕业设计（论文）

基于 Android 的 B2B 电影电商平台
设计与实现

**Design and Implementation of Movie E-Commerce
Platform for Business-to-Business Based on Android**

学 院：____软件学院____

专 业：____软件工程____

学生姓名：____XXX____

学 号：____XXX____

指导教师：____XXX____

北京交通大学

2016 年 5 月

学士论文版权使用授权书

本学士论文作者完全了解北京交通大学有关保留、使用学士论文的规定。特授权北京交通大学可以将学士论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：

指导教师签名：

签字日期： 年 月 日

签字日期： 年 月 日

中文摘要

由于我国电影产业各个环节中存在严重的资源不对等、信息不对称，利益链条无法达到充分整合，产业内互斥性竞争现象普遍存在。同时，由于机制的不健全，对知识产权的保护不充分，导致市场中盗版横行，使得衍生品等发展陷入困局。本平台瞄准的是当前国内电影产业的痛处，希望通过这个产品将制片、院线等产业链中的各环节有机整合起来，并建立健康的全产业链生态圈，进而促进以衍生品市场为核心的电影产业生态圈的良性发展，同时也进一步帮国内电影产业缩小与国外的产业差距。

本系统是一款针对片商、影院和院线等电影行业人士的 B2B 服务产品，主要分为首页展示、票房分析、正品采购、行业资讯、商务订单以及个人中心六大模块。首页展示模块主要包含来自票房分析、正品采购及行业资讯的精选内容。票房分析模块基于专业知识及用户的海量数据分析，全面分析电影排片和票房数据，涵盖影片、影院、院线的实时/历史票房查询、实时排片多维度的对比分析功能。正品采购模块提供企业级一站式正版电影衍生品解决方案，商品种类最全，价格实惠，覆盖全国 400 多个城市配送和物流实时跟踪。行业资讯模块第一时间报道产业资讯。商务订单模块包含商务用户所负责的订单显示及相关操作。个人中心模块包含账户绑定，我的订单，我的账号等账号相关信息。该系统完全基于 Android 平台进行开发。

目前，该系统已经正式投放使用，为多家院线、影院和片商等电影行业人士提供了高质量的服务，同时也为公司带来了可观的收益。

关键词：B2B；电影电商；Android

ABSTRACT

Because of the serious asymmetric of information and resources of our country, the interests of the chain can not be fully integrated, competition phenomenon exists widely in industry. At the same time, because the mechanism is not integrity, the protection of intellectual property is not sufficient, leading to rampant piracy in the market, making the development of derivatives into a dilemma. This platform aimed at the weakness of the domestic film industry, wish to integration Producer, Cinema and other industry chain Organically through this product, and establish a healthy Industrial chain ecosystem, and promote the development of the Industrial chain ecosystem which use the derivatives market as core. Also, it can help deduce the gap between domestic film and foreign industry.

This system is a B2B service product be aimed at film makers, theaters and cinemas who have relationship with film industry, mainly for home page display, box office analysis, genuine procurement, industry information, business orders as well as personal center six modules. Home page display contains the selected content from the box office analysis, genuine purchasing and industry information. Box office analysis based on the professional knowledge and the vast data analysis of users, analysis the movie film and box office data comprehensively, include the film, theater, cinema real-time /past times box office query, comparison of the realtime sheets in a row multi dimension analysis function. Genuine purchasing module provides enterprise class one-stop genuine solutions of film derivative, the types of goods is the most comprehensive and it's affordable, covering more than 400 cities In the whole country, Distribution and logistics is real time tracking. Industry information module report the industry information for the first time. The business order module includes the order display and the related operation of the business users. Individual center module contains account binding, my order, my account and other information with account number. The system is developed on the Android platform entirely.

At present, the system have put into use already, provide the high quality service of many theaters, cinemas and film makers such as the film industry, and it also bring a Substantial gains for the company.

KEYWORDS: Business-to-Business; Movie E-Commerce; Android

目 录

中文摘要.....	I
ABSTRACT	II
目 录	III
1 引言	1
1.1 项目的背景及意义	1
1.1.1 项目背景	1
1.1.2 项目意义	2
1.2 项目难点及解决思路	2
1.2.1 项目难点	2
1.2.2 解决思路	3
1.3 关键任务及交付	3
1.4 论文的组织结构	4
1.5 本章小结	4
2 相关技术介绍	5
2.1 网络通信框架——VOLLEY	5
2.2 图片处理框架——FRESCO	5
2.3 数据交换格式 JSON 与 FASTJSON	6
2.4 内存泄露检测工具——LEAKCANARY	6
2.5 集成开发环境——ANDROID STUDIO	7
2.6 版本控制工具——GIT	7
2.7 本章小结	8
3 系统需求分析	9
3.1 系统整体功能概述	9
3.2 系统功能性需求分析	9
3.2.1 首页展示模块	9
3.2.2 票房分析模块	10
3.2.3 正品采购模块	11
3.2.4 行业资讯模块	11
3.2.5 商务订单模块	12
3.2.6 个人中心模块	13
3.3 系统的非功能性需求	14
3.4 本章小结	14
4 系统概要设计	15
4.1 客户端整体架构设计	15
4.2 客户端功能模块划分	17

4.3 客户端数据结构设计	17
4.4 本章小结	18
5 系统详细设计与实现	19
5.1 首页导航的设计与实现	19
5.2 首页展示模块的设计与实现	20
5.2.1 BannerView	22
5.2.2 HomeTicketView	22
5.2.3 HomeMallView	23
5.2.4 HomeNewsView	24
5.2.5 CaptureActivity	25
5.3 票房分析模块的设计与实现	25
5.4 正品采购模块的设计与实现	27
5.4.1 TabMallFragment	27
5.4.2 商城列表	29
5.4.3 购物车	30
5.4.4 提交订单	30
5.4.5 商品详情	32
5.5 行业资讯模块的设计与实现	32
5.6 个人中心模块的设计与实现	34
5.7 商务订单模块的设计与实现	37
5.8 本章小结	39
6 系统测试	40
6.1 功能性测试用例设计	40
6.2 性能测试用例设计	41
6.3 本章小结	42
7 总结与展望	43
参考文献	44
致 谢	45
附 录	46

1 引言

本章为引言部分，主要介绍基于 Android 的 B2B 电影电商平台的项目背景、意义、开发中的难点及对应的解决思路，作者在开发中主要的工作，同时对笔者在项目中承担的主要工作和产出进行了介绍。

1.1 项目的背景及意义

本节将介绍项目的背景与意义，主要介绍该系统产生的背景、行业现状、该系统的用户人群、该系统的适用范围以及该项目的开发意义。

1.1.1 项目背景

据了解，以美国为首的国外电影产业早已形成机制健全、信息充分流通的产业链生态圈，其行业利润不仅仅依靠票房，而是通过衍生品市场进入了新的高度。有数据显示，6 部《星球大战》全球票房收入 45 亿美元左右，但其衍生品收入已超过 200 亿美元。

但另一方面，由于我国电影产业各个环节中存在严重的资源不对等、信息不对称，利益链条无法达到充分整合，产业内互斥性竞争现象普遍存在。同时，由于机制的不健全，对知识产权的保护不充分，导致市场中盗版横行，使得衍生品等发展陷入困局。

本平台瞄准的是当前国内电影产业的痛处，希望通过这个产品将制片、院线等产业链中的各环节有机整合起来，并建立健康的全产业链生态圈，进而促进以衍生品市场为核心的电影产业生态圈的良性发展，同时也进一步帮国内电影产业缩小与国外的产业差距。

本系统目标是打造一站式的衍生品 B2B 电商平台，从预览、送品、支付到跟踪物流配送，为产业链中的衍生品经销合作伙伴提供全套的解决方案。这不仅是单纯的采购服务，还会有专业权威的数据和咨询服务为企业提供参考和指导，更重要的是，本系统完全通过移动互联网来搭载，方便企业及时、灵活掌握衍生品采购和产业的一切动态。

作为国内首屈一指的电影互联网内容平台，时光网拥有丰富而权威的电影产业资讯，商户能通过本系统及时了解到行业内的最新风向，从而有效把握业内最全讯息和最新潮流。资讯中不乏时光网自主原创内容，使得本系统能够成为公开透明的电影产业信息交流平台。同时，本系统还了解到产业人士对于相关公众号关注阅读的需求，聚合了业内最权威的公众号，某种程度上成了行业优质内容的“联盟”，精准地为使用者提供有效信息。

1.1.2 项目意义

业内人士认为，其实国内已经存在各种衍生品采购平台，不过由于所提供的服务过于单一，只注重衍生品的采购环节，其用户规模得不到充分释放，本系统依托于时光网现有的成熟业务体系和十年来积累的丰富产业链资源，有实力渗透、打通产业链上下游，在为商业伙伴提供采购渠道的同时更能为他们提供增值服务，从而让衍生品的采购和营销决策更精明合理。

值得一提的是，时光网作为国内唯一拥有好莱坞六大片商授权的平台，同时也是国内被授权品类最全面的被授权方，掌握着大量热门 IP 资源，加之时光网目前独步国内的衍生品独立自主设计、生产能力的 **Designed by Mtime**，时光网所能提供的衍生品阵容可以说是国内最丰富的。再考虑到本系统提供的保姆式服务，这一平台完全有理由成为企业采购衍生品的不二之选。

总体而言，本系统是目前国内少有的一款优质的电影产业专业服务产品，作为一个正版电影衍生品 **B2B** 移动电商平台，提供的不仅是单纯的商业服务，更提供给商户一个全产业链视角的生态化的商务平台。

此外，时光网获得的授权目前在国内因为难有出其右者，加上业内一流的独立设计与研发能力，使其能够源源不断地输出优质的正版电影衍生品。

如此一来，商户在经营衍生品的过程中，除了拥有一个品质有保障的正版衍生品供应渠道，还能有效掌握来自于电影后产品市场的上下游动态，及时调整产品策略，从而更好地进行衍生品的经营。

1.2 项目难点及解决思路

本节主要介绍该系统在设计和开发过程中的难点，以及所对应的解决思路。

1.2.1 项目难点

该项目的难点主要如下：

（1） 该项目存在大多数页面都会加载大量高清图片，而 **Android** 系统对单个应用的运行时内存有着严格的限制，我们必须防止因图片处理而引起的内存溢出，同时又要设置合理的图片缓存结构，让其使用起来足够流畅。

（2） 该项目的数据基本都是从远程服务器获取，部分情况下一个页面需要大量数据获取，因此必须保证网络通信足够迅速且稳定，同时数据也能安全传输。

（3） 由于移动平台网络的特殊性，网络环境可能得不到保证，尤其是在 **2g/3g** 网

络情况下，网络传输速率会很低，因此需要尽可能的使用轻量级的数据格式。

（4） 系统部分页面使用 Html 5 开发，必须保证原生页与 Html 5 页能够顺利且流畅的进行交互。

1.2.2 解决思路

对应于上述难点，该系统得出了极佳的解决思路，其解决方案如下，其中具体的技术方案将在下一章进行详细介绍。

（1） 图片处理采用了 Fresco 图片处理框架，在图片缓存方面拥有三级缓存：Bitmap 缓存+未解码图片缓存+硬盘缓存，有效的解决了内存溢出问题，同时又能让用户流畅的加载图片。

（2） 网络通信采用了 Volley 网络通信框架，通过多个并发请求队列，实现轻松加载网络上的图片。

（3） 后台传输采用了轻量级的 Json 数据交换格式，其数据体积较小，易于传输，同时客户端采用了 FastJson 框架，更快的对服务器返回的 Json 数据格式进行解析，使数据传输解析速率达到最高。

（4） 项目中采用了与 Html 5 交互的 JSSdk，将所需的与 Html 5 页面交互的规则定制到其中，有效地保证原生页与 Html 5 页能够顺利且流畅的进行交互。

1.3 关键任务及交付

作者参与整个项目的开发过程，并在项目上线后负责对系统进行维护，开发阶段的主要工作为：

- （1） 参与整个系统的架构设计。
- （2） 个人中心模块的开发。着重解决不同类型用户的商品订单管理页。
- （3） 商务订单模块的开发。主要对订单详情和订单筛选进行开发。
- （4） 行业资讯模块的开发。设计不同资讯类型的显示与交互。
- （5） 正品采购模块的开发。主要负责其中的购物车模块。

在项目上线后，作者主要进行的工作：

- （1） 监测与维护项目的运行。
- （2） 对负责模块进行迭代开发。
- （3） 编写自动化测试代码。

1.4 论文的组织结构

本论文共分为七章，具体内容安排如下：

第一章为绪论。介绍了项目的背景与意义，项目的重点及特色，阐明了本人在毕业设计中的主要工作，并且说明了论文的组织结构。

第二章介绍了毕业设计中用到的相关技术，包括 Volley 网络通信框架、Fresco 图片处理框架、Android Studio、Git、LeakCanary、Json 与 FastJson。

第三章介绍了系统的需求分析。主要包含系统的整体需求和各模块的功能需求，以及系统的非功能性需求。

第四章介绍了系统的概要设计。主要包括系统的整体架构设计，各个功能模块之间的划分与功能模块包含的重要功能，还对系统所使用的数据结构进行介绍，通过表格的形式列举出部分重点数据。

第五章介绍了系统的详细设计与实现。主要包括系统各个模块的设计与实现。

第六章介绍了系统的测试。主要通过 Robotium 自动化测试框架编写自动化测试脚本。

第七章是总结与展望，说明了系统在未来可以改进的方向，对系统的发展前景做出了展望。

1.5 本章小结

本章介绍了项目的背景与意义，说明了项目的难点及对应的解决思路，对笔者在项目中承担的主要工作和产出进行了介绍，并且对论文的组织结构进行了详细的分析。

2 相关技术介绍

本章主要介绍笔者在毕业设计中用到的一系列技术，通过比较分析得出最终的技术方案，包括网络通信框架 Volley，图片处理框架 Fresco，json 数据与 FastJson 框架，内存检测工具 LeakCanary，版本控制系统 GIT，IDE Android Studio。

2.1 网络通信框架——Volley

Android 开发中，绝大部分网络通信过程中都会用到 HTTP(超文本传送协议)协议来进行网络交互。Android 平台提供的 SDK(软件开发工具包)中主要提供两种方法来进行 HTTP 通信——HttpClient 和 HttpURLConnection。但当应用比较复杂的时候需要我们编写大量的代码，如果封装不当，将会造成极大的资源浪费^[5]。

不过 HttpURLConnection 和 HttpClient 的用法还是稍微有些复杂的，如果不进行适当封装的话，很容易就会写出不少重复代码。于是乎，一些 Android 网络通信框架也就应运而生，比如说 AsyncHttpClient，它把 HTTP 所有的通信细节全部封装在了内部，我们只需要简单调用几行代码就可以完成通信操作了。再比如 Universal-Image-Loader，它使得在界面上显示网络图片的操作变得极度简单，开发者不用关心如何从网络上获取图片，也不用关心开启线程、回收图片资源等细节，Universal-Image-Loader 已经把一切都做好了^[9]。

Android 开发团队也是意识到了有必要将 HTTP 的通信操作再进行简单化，于是在 2013 年 Google I/O 大会上推出了一个新的网络通信框架——Volley。Volley 可是说是把 AsyncHttpClient 和 Universal-Image-Loader 的优点集于了一身，既可以像 AsyncHttpClient 一样非常简单地进行 HTTP 通信，也可以像 Universal-Image-Loader 一样轻松加载网络上的图片。

Volley 的名称由来：A burst or emission of many things or a large amount at once，因此其特点也是特别适合数据量不大但是通信频繁的场景，这基本也是本系统的使用场景。

2.2 图片处理框架——Fresco

在 Android 设备上面，快速高效的显示图片是极为重要的。过去的几年里，我们在如何高效的存储图像这方面遇到了很多问题。图片太大，但是手机的内存却很小。每一个像素的 R、G、B 和 alpha 通道总共要占用 4byte 的空间。如果手机的屏幕是 480*800，那么一张屏幕大小的图片就要占用 1.5M 的内存。手机的内存通常很小，特别是 Android 设备还要给各个应用分配内存。在某些设备上，分给你的应用内存仅仅有 16MB。一张

图片就要占据其内存的十分之一。所以，太多的图片总会耗去大部分的内存，从而导致应用的崩溃。

为了解决这一问题，Facebook 2015 年推出了一款用于 Android 应用中展示图片的强大图片库 Fresco，它能够从网络、本地存储和本地资源中加载图片。而且，为了节省数据和 CPU，它拥有三级缓存：Bitmap 缓存+未解码图片缓存+硬盘缓存。此外，Fresco 在显示方面是用了 Drawees，这是一个像 MVC 架构的图片显示框架。该模型被称为 DraweeHierarchy。它被实现为 Drawables 的一个层，对于底层的图像而言，每一个曾都有特定的功能——成像、层叠、渐变或者是放缩。它可以显示占位符，直到图片加载完成。而当图片从屏幕上消失时，会自动释放图片所占的内存。同时还支持 webp 格式的图片，是 Google 官方推行的，它的大小比其它格式图片的大小要小一半左右，目前各个大公司都渐入的使用这种图片格式了。Fresco 支持 Android 2.3 及以上版本。

2.3 数据交换格式 Json 与 Fastjson

JSON 是 JavaScript 对象表示法，JavaScript Object Notation。是一种轻量级的数据交换格式，具有良好的可读和便于快速编写的特性。JSON 采用完全独立于语言的文本格式，是理想的数据交换语言。易于人阅读和编写，同时也易于机器解析和生成。它独立于语言，具有自我描述性，更加易于理解。JSON 使用 JavaScript 语法来描述数据对象，但是 JSON 仍然独立于语言与平台^[1]。JSON 解析器和 JSON 库支持许多不同的编程语言。正是由于 json 数据的这些特性，使它成为现在 Android 开发的主流数据交换格式。

Fastjson 是由阿里巴巴开发的一个 Java 语言编写的高性能功能完善的 JSON 库。它采用一种“假定有序快速匹配”的算法，把 JSON Parse 的性能提升到极致，是目前 Java 语言中最快的 JSON 库。测试表明，Fastjson 具有极快的性能，超越任其他的 Java Json parser，包括自称最快的 JackJson。功能强大，完全支持 Java Bean、集合、Map、日期、Enum，支持范型，支持自省，无依赖，能够直接运行在 Java SE 5.0 以上版本。已经被广泛使用在缓存序列化、协议交互、Web 输出、Android 客户端等多种应用场景。

Fastjson 提供 Android 版本，和标准版本相比，Android 版本去掉一些 Android 虚拟机 dalvik 不支持的功能，使得 jar 更小，同时针对 dalvik 做了很多性能优化，包括减少方法调用等。

2.4 内存泄露检测工具——LeakCanary

LeakCanary 是由 Square 开发的一个内存泄露自动探测神器，它是一个 Android 和 Java 的内存泄露检测库，可以大幅度减少了开发中遇到的 OOM 问题。其使用简单，直

接通过通知框的方式让开发者获取自己应用的内存泄露情况。而且得益于 gradle 强大的可配置性，可以确保只在编译 debug 版本时才会检查内存泄露，而编译 release 等版本的时候则会自动跳过检查，避免影响性能。

Square 有自己的库去解析 heap dump 文件，而且实现的很容易。通过打开 heap dump，加载进来，然后解析。然后根据 key 找到我们的引用。然后根据已有的 Key 去查看拥有的引用。拿到实例，然后得到对象图，再反向推导发现泄漏的引用。

所有的工作实际上都发生在 Android 设备上。当 LeakCanary 探测到一个 Activity 已经被销毁掉，而没有被垃圾回收器回收掉的时候，它就会强制导出一份 heap dump 文件存在磁盘上。然后开启另外一个进程去分析这个文件得到内存泄漏的结果。如果在同一进程做这件事的话，可能会在尝试分析堆内存结构的时候而发生内存不足的问题。

最后，你会得到一个通知，点击一下就会展示出详细的内存泄漏链。而且还会展示出内存泄漏的大小，你也会很明确自己解决掉这个内存泄漏后到底能够解救多少内存出来。LeakCanary 也是支持 API 的，这样你就可以挂载内存泄漏的回调，比方说可以把内存泄漏问题传到服务器上，在测试阶段出现内存泄漏的时候，它就会发出通知。

2.5 集成开发环境——Android Studio

Android Studio 是一个为 Android 平台开发程序的集成开发环境。2013 年 5 月 16 日由 Google 产品经理 Ellie Powers 在 Google I/O 上发布，可供开发者免费使用。

Android Studio 基于 JetBrains IntelliJ IDEA，为 Android 开发特殊定制，提供了集成的 Android 开发工具用于开发和调试，并在 Windows、Mac OS X 和 Linux 平台上均可运行。在 IDEA 的基础上，Android Studio 提供基于 Gradle 的构建支持，Android 专属的重构和快速修复，提示工具以捕获性能、可用性、版本兼容性问题，支持 ProGuard 和应用签名，基于模板的向导来生成常用的 Android 应用设计和组件，功能强大的布局编辑器，可以让你拖拉 UI 控件并进行效果预览。Android 官方已经选定 Android Studio 为其官方开发工具，并且已经不再支持 Eclipse ADT 的更新。

在几个月前的 Android 开发者大会中，Google 高调发布了 Android Studio 2.0。Android Studio 2.0 的主题是提速，新特性主要包括代码热更新（Instant Run）、GPU 分析工具、Gradle 速度提升、高性能模拟器等。强大的功能使 Android Studio 已经成为时下最热门的 Android 平台集成开发环境。

2.6 版本控制工具——GIT

GIT 是用于 Linux 内核开发的版本控制工具。与 CVS、Subversion 一类的集中式版

本控制工具不同，它采用了分布式版本库的作法，不需要服务器端软件，就可以运作版本控制，使得源代码的发布和交流极其方便。GIT 的速度很快，这对于诸如 Linux 内核这样的大项目来说自然很重要。GIT 最为出色的是它的合并追踪（merge tracing）能力^[3]。

实际上内核开发团队决定开始开发和使用 GIT 作为内核开发的版本控制系统的时候，世界上开源社区的反对声音不少，最大的理由是 GIT 太艰涩难懂，从 GIT 的内部工作机制来说，的确是这样。但是随着开发的深入，GIT 的正常使用都由一些友善的命令稿来执行，使 GIT 变得非常好用。现在，越来越多的著名项目采用 GIT 来管理项目开发，例如：wine、U-boot 等。

2.7 本章小结

本章主要介绍了客户端在开发过程中使用的关键技术。客户端主要采用 Java 语言开发，用到了 Volley、Fresco 等大型框架，通过使用这些关键技术，系统在性能上将完全满足最低使用标准。

3 系统需求分析

本章将通过分析系统的功能得到系统的整体需求，并对系统的工作流程进行说明。然后将根据系统的整体业务流程划分系统的功能模块，并对系统的各个模块进行详细的功能性需求分析，最后将对系统整体的非功能性需求进行分析。

3.1 系统整体功能概述

本系统主要分为首页展示、票房分析、正品采购、行业资讯、商务订单以及个人中心六大模块。首页展示模块主要包括广告展示、票房展示、商品展示、资讯展示等功能。票房分析模块主要包括今日上映、院线票房统计、影院票房统计、上映日历、票房对比、影院影片院线搜索等功能。正品采购模块主要包括商品列表、商品推荐、商品搜索、商品详情、电影主题商品、购物车以及支付功能。行业资讯模块主要包括资讯列表、资讯分类、资讯详情、资讯分享等功能。商务订单模块主要包括订单详情、订单列表、订单筛选、修改订单等功能。个人中心模块主要包括查看订单、登录注册、账户管理、密码修改、帮助中心等功能。

3.2 系统功能性需求分析

根据系统的整体功能和业务流程，将该系统分为首页展示、票房分析、正品采购、行业资讯、商务订单以及个人中心六大模块。本节将依次介绍各模块的功能性需求。

3.2.1 首页展示模块

首页展示是用户进入系统后看到的第一个页，因此主要包含广告展示、票房展示、商品展示、资讯展示等展示功能，将最核心的内容第一时间展现给用户，引导用户快速了解本系统，同时包含 TitleBar 跳转功能，方便用户快速跳转，以下是详细的功能描述。

表 3-1 首页展示模块功能描述

功能模块	功能	功能描述
首页展示模块	广告 Banner 展示	通过一个可滑动的 Banner 进行广告展示，Banner 内容及条数等数据通过接口返回进行动态展示，可以通过点击具体广告跳转到指定的页。

表 3-1 续 首页展示模块功能描述

功能模块	功能	功能描述
首页展示模块	票房分析展示	显示当日全国总票房，拍片数量以及新上映三条数据，数据类型可能根据接口返回进行动态更改。同时显示当日前三的电影详情，点击更多跳转到票房分析模块。
	正版采购展示	显示两个热门电影的主题周边推荐，可以点击更多商品跳转到该电影主题周边详情页，同时也可点击更多跳转到正版采购模块。
	行业资讯展示	精选三条当天最热门资讯精选展示，同时可以跳转快讯、声音、解读、产业圈四个资讯子模块，也能点击更多跳转至行业资讯模块。
	Titlebar 跳转	Titlebar 包含二维码扫描和购物车，二维码扫描可以扫描商品二维码等进行操作，购物车会显示当前购物车商品数量，点击后跳转至购物车详情页。

3.2.2 票房分析模块

票房分析模块基于专业及用户的海量数据进行分析，全面分析电影排片和票房数据，涵盖影片、影院、院线的实时/历史票房查询、实时排片多维度的对比分析功能。以下是详细的功能描述。

表 3-2 票房分析模块功能描述

功能模块	功能	功能描述
票房分析模块	实时票房	包含当日的总票房，总排片以及当日票房前十的影片的具体数据展示，并通过一个扇形图展示排片比例，此外还可查询历史票房。
	实时排片	以排片场次为基准显示前十的影片数据展示，同时支持收藏影片的具体数据展示。
	上映日历	通过一个列表按上映日期线展示出从今天开始的即将上映影片展示，同时支持在日历上按日、月、年选择日期来展示上映列表。
	对比分析	精选三条当天最热门资讯精选展示，同时可以跳转快讯、声音、解读、产业圈四个资讯子模块。
	影片/院线/影院搜索页	顶部包含一个对影片/院线/影院信息进行搜索的搜索框，支持影片分类筛选和影院分类筛选，同时还有热门影片/院线/影院推荐。

3.2.3 正品采购模块

正品采购模块提供企业级一站式正版电影衍生品解决方案，商品种类最全，价格实惠，覆盖全国 400 多个城市配送和物流实时跟踪。主要包括商城首页、商品列表、商品搜索、商品详情、购物车以及支付功能。以下是详细的功能描述。

表 3-3 正品采购模块功能描述

功能模块	功能	功能描述
正品采购模块	商城首页	点击采购 Tab 后所展示的页，包含顶部商品推荐 Banner 以及多个热销商品推荐位及以电影为主题的商品推荐位，可以在后台实时更改推荐的商品内容，让用户在首页便能看到当前最热门的电影衍生品。
	商品列表	商品列表页，每一个商品展示一张商品图及少量商品信息，支持价格排序，价格区间、玩具模型、数码周边等多维度筛选，还可根据影片筛选出影片周边。
	商品搜索	通过一个顶部搜索框进行商品搜索，支持相关电影，商品名称，商品描述等多维度模糊查询。
	商品详情	商品详情包括多张商品图片，商品价格，商品图文详情及规格参数。通过选择 sku 及数量进行购物车添加，点击右上角分享可将商品信息分享到微信、微博等主流社交平台。
	购物车	通过一个列表显示已添加至购物车的 sku 信息，可以对 sku 进行收藏、移除购物车、数量修改、sku 选择结算操作。
	订单填写及支付	展示订单的 sku、商品总价、运费、结算价格，用户可以选择收货地址、发票信息、送货时间，添加优惠券。订单提交后可以选择支付宝、微信及银联支付。

3.2.4 行业资讯模块

行业资讯第一时间报道产业资讯，最全角度解读票房现象，深入观察探寻业内风向，除丰富的原创外，汇聚相关公众号优质内容，一网打尽行业所有好料。主要包括快讯、声音，解读与产业圈四个模块，第一时间提供行业动态，倾听行业资深人士的评价与分析，以下是详细的功能描述。

表 3-4 行业资讯模块功能描述

功能模块	功能	功能描述
行业资讯模块	资讯列表	资讯列表通过内容分为全部、快讯、声音、解读、产业圈五个模块，每条新闻包含标题、类型、图片、时间、简介，让用户能够通过列表浏览新闻内容。
	资讯详情	通过点击资讯列表进入资讯详情，资讯分为图片资讯、视频资讯、文字资讯以及图文资讯四大类型，针对每种类型资讯定制了专属的模板进行展示。
	资讯分享	通过点击顶部栏分享按钮进行分享，可以分享到微信、微博、QQ 等主流社交软件，并对每种类型资讯定制了不同的分享模板。

3.2.5 商务订单模块

商务订单模块用于商务账号(商务账号用户是公司对采购订单进行管理的专职人员)对负责订单进行查询管理，主要包括订单列表、订单详情、订单筛选等功能，以下是详细的功能描述。

表 3-5 商务订单模块功能描述

功能模块	功能	功能描述
订单模块	订单列表	以列表的形式展示当前商务账号的所有订单，每一项包含一个订单的基本信息，同时提供订单筛选功能。
	订单详情	订单详情包含订单的详细信息，包括购买商品数量、价格等，支持修改 sku 价格，可以进行发票类型修改，订单审核等操作。
	订单搜索及筛选	对订单支持公司名称、下单人、订单编号及商品名称进行模糊搜索，同时支持区域名称、集团名称、支付状态、发票状态等多种条件进行筛选。
	订单跟踪	显示订单的基本信息及订单状态，同时显示每个商品的基本信息及状态，包含配送中，已收货等多种状态，对于配送中商品可以查看物流，对于整个订单可以点击确认收货按钮。
	发票查询	发票查询通过列表的形式显示每个商品的发票详情，对选择发票配送的发票可以查看配送物流。

3.2.6 个人中心模块

个人中心模块用于用户进行账户相关的信息查询管理，主要包括查看订单、登录注册、账户管理、帮助中心等功能，以下是详细的功能描述及用例图。

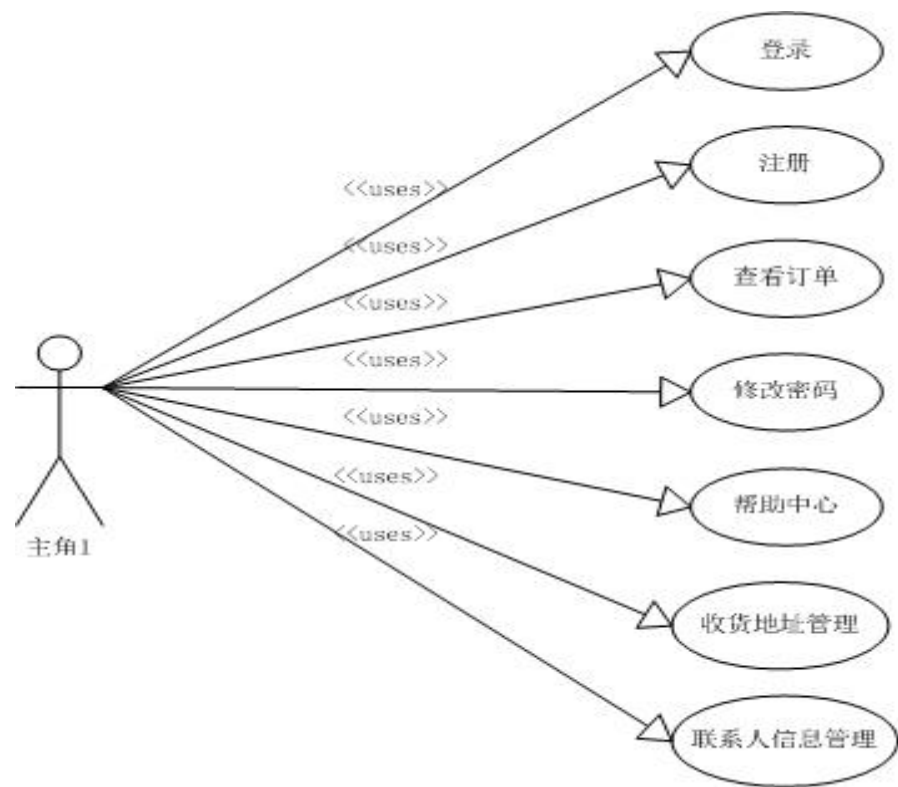


图 3-1 用例描述

表 3-6 个人中心模块功能描述

功能模块	功能	功能描述
个人中心模块	查看订单	可以进行本账号订单的查询，以列表的形式展示，可以点击查看订单详情。
	登录注册	支持企业客户注册，仅提供手机号注册，注册成功后需补充填写公司及个人真实信息，信息补充完毕后方可使用账号密码进行登录。
	账户管理	账户管理主要包括联系人信息管理、收货地址管理、修改密码三大功能，用户可以对联系人和收货地址进行增删改查，同时支持密码修改。
	帮助中心	用于帮助用户快速了解本系统，包括购物指南、支付指南、配送方式、售后服务、服务条款等多种选项。

3.3 系统的非功能性需求

非功能性需求，是指软件产品为满足用户业务需求而必须具有且除功能需求以外的特性。软件产品非功能性定义不仅决定产品的质量，还在很大程度上影响产品的功能需求定义。结合该系统的功能特点，该系统需要具有以下非功能性需求，如表 3-7 所示。

表 3-7 非功能性需求

非功能性需求	详细要求
安全性	用户信息统一调用 UserInfo 接口来获取，传输过程采用 32 位 MD5 算法进行加密处理，极大程度的保证了用户信息的安全性。
可靠性	系统是在 GIT 进行管理开发，系统数据实时备份，同时存储到多台本地服务器与云服务器，即使某台服务器出现故障，也能很快切换到其他备份服务器上，以最快的时间恢复服务。
稳定性	系统有异常统计服务，当客户使用中出现异常时会将异常日志发送到服务器，每天对异常进行分析后优化系统，使系统 crash 率能够控制在 0.05%以内。
可用性	由于移动平台的特性，本系统全年可用。
易用性	页面开发符合 Android 开发规范，用户可以轻松上手，同时配有用户帮助模块和在线客户帮助用户解决使用中出现的的问题。
可扩展性	系统需求定位明确，开发初期即决定采用敏捷开发，因此现有框架能够轻松的将新业务接入现有系统。

3.4 本章小结

本章首先简单介绍了系统的主要功能，接着从功能性需求和分功能性需求两方面对系统进行详细描述。功能性需求通过将系统功能模块进行明确划分，利于任务与职责分配，从而更好的进行模块迭代开发。其次将每个模块的重点功能与内容列举出来进行简要介绍，方便开发人员迅速了解自己的主要工作。本章也对系统所应该具有的非功能性需求进行了分析，包括安全性、稳定性、可靠性、可用性、易用性和可扩展性。这些非功能特性保证了系统的质量。

4 系统概要设计

本章将根据需求分析的结果对整个客户端进行概要。系统概要设计的内容分为客户端整体架构的设计、客户端功能模块划分及数据库设计。

4.1 客户端整体架构设计

该客户端在架构设计上采用 MVC(Model-View-Controller, 模型-视图-控制器)框架, 该框架将代码分为模型、视图和控制器三层, 视图层和业务层分离, 这样就允许更改视图层代码而不用重新编译模型和控制器代码, 同样, 一个应用的业务流程或者业务规则的改变只需要改动 MVC 的模型层即可。因为模型与控制器和视图相分离, 所以很容易改变应用程序的数据层和业务规则。其设计架构如图 4-1 所示, 除了 MVC 架构外还添加了数据储存和外部服务的表示。

- 1) 模型层: 模型层为客户端用于处理应用程序数据逻辑 UI 显示的部分, 包括网络数据访问与写入、本地缓存文件读取与写入、本地数据库的操作以及数据加工转换等主要功能。
- 2) 视图层: 视图层为客户端 UI 显示, 主要通过 xml 文件进行界面描述, 部分控件采取代码自定义的形式进行绘制。本层的显示数据来源于模型层, 同时也可以获取用户输入为模型层处理加工数据做准备。
- 3) 控制层: 控制层主要负责客户端逻辑处理, 从模型层获取数据, 将其处理后展示给用户, 在模型层与视图层中起到桥梁的作用, 该客户端控制层的职责主要由 Activity 和 Fragment 进行承担。
- 4) 数据存储层: 数据储存层用于用户信息存储, 主要通过网络云存储, 用户通过网络访问的方式获取和写入数据, 同时提供文件缓存、本地数据库 SQLite 和 SharedPreferences 进行辅助存储, 对于不同业务需求及数据量大小的不同, 将会采取最优的存储方式。
- 5) 外部服务: 该客户端除了本地开发外还采用了不少外部框架进行辅助开发, 主要包括 Volley、Fresco 和 FastJson 等, 具体功能参见第二章。
- 6) 如图 4-1 所示, 系统由上至下进行请求数据, view 层获取用户操作, 将操作传递至 Controller 层, Controller 层根据不同操作进行不同 Model 加载, Model 层根据数据类型进行相应数据请求, 数据存储层完成数据加载后又由下至上进行数据传输, 由数据存储获取数据填充到 Model 层, Model 层获取数据后传递至 Controller 层, Controller 将数据再用于填充布局, 最后通过 view 层进行视图交互。

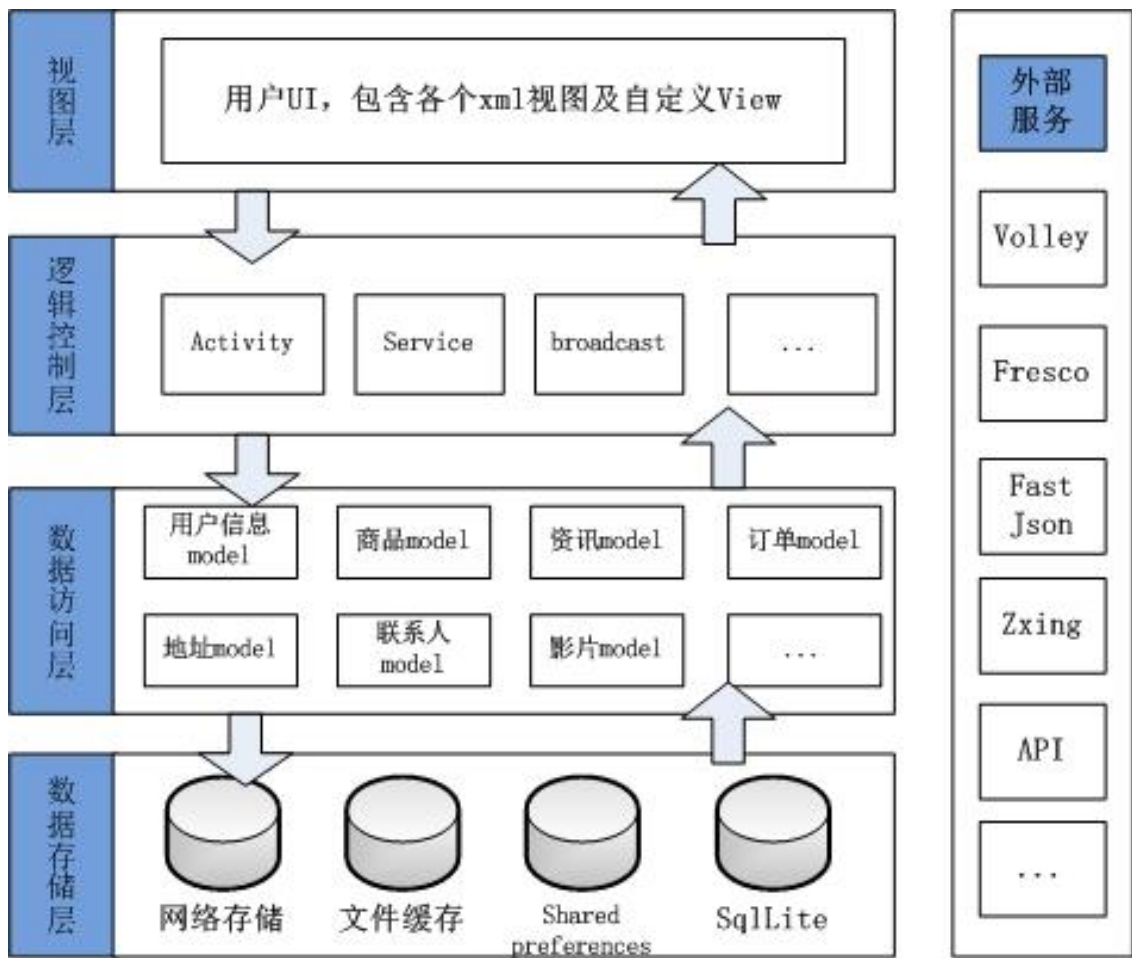


图 4-1 客户端架构设计图

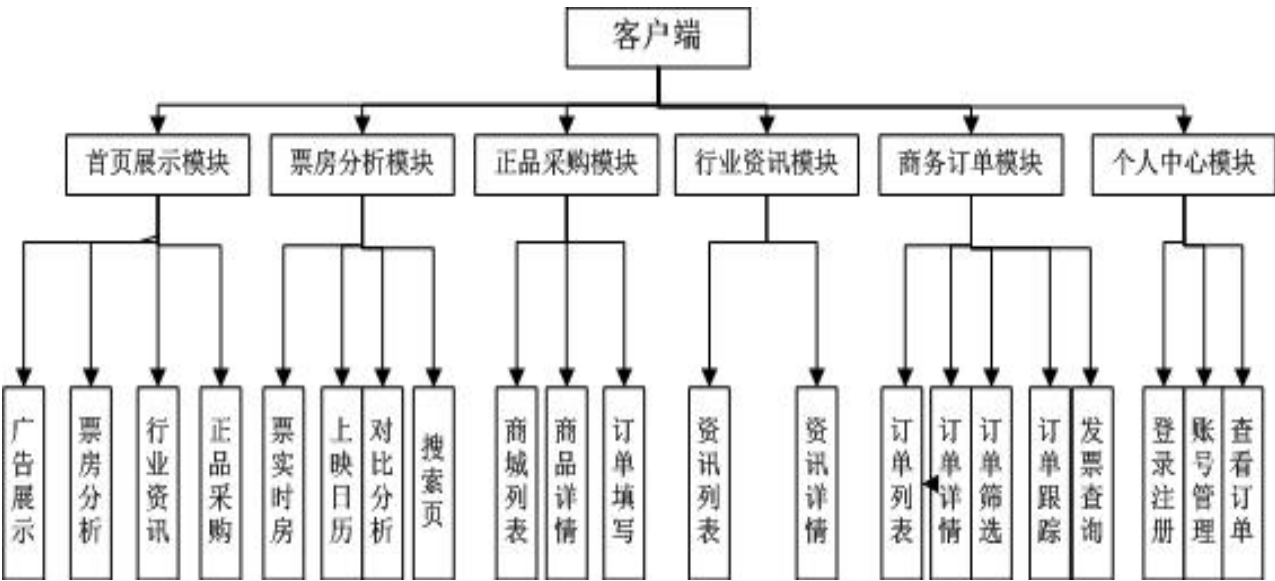


图 4-2 功能模块组织结构

4.2 客户端功能模块划分

根据客户端的总体功能性需求分析得到 6 个功能相互独立的功能模块，分别为首页展示、票房分析、正品采购、行业资讯、商务订单以及个人中心模块，除商务订单模块外每个模块的入口页均为主页上的一个 Tab。六个模块均具备自己独立的功能，具体功能模块组织结构如图 4-2 所示。

4.3 客户端数据结构设计

根据客户端的总体功能性需求分析进行数据结构设计，该客户端所有数据均以 Json 格式进行传输，因此客户端只需设计自己所需的数据格式即可，后台人员再根据现有逻辑和客户端所需数据进行整合，最后得到最终的数据结构传递给客户端，与用户相关联的数据表结构图如图 4-3 所示，下面将对其中表进行简要描述。

用户信息数据结构如表 User 所示。userId 用于标识具体用户，userType 为用户类型，目前支持商务用户和普通用户，userLevel 为用户等级，不同用户等级权限不同，groupId 为所在区域 id，roles 为联系人列表，merchantId 为所属商务 Id，addressId 为收货地址 Id。

用户订单数据结构如 Order 表所示，一个用户可以具有多个订单，其中 orderId 唯一标识具体订单，txtAmount 为总金额，dedutedAmount 为优惠金额，saleAmount 为订单实付金额，invoiceTitle、invoiceType 分别为发票抬头与类型，无发票时可以为空，paymentMode、paymentWay 分别为支付模式和支付方式。

商品数据结构如表 Goods 所示，其中 minPrice 为会员折扣后的价格，backGround 为商品背景介绍，image 为商品图片 url 集合，一个商品可以具有多张商品图片，一个订单可以包含多件商品。

地址数据结构如 Address 表所示，postCode 为邮政编码，除它之外均为必填选项，其中 cityId、provinceId、districtId 分别为所在城市，省份和区域 Id，一个公司对应一个地址，一个用户可以具有多个收货地址。

公司数据结构如 Company 表所示，addressId 对应其详细地址，time 为注册时间，owner 为公司注册者，image 为公司 logo 图片 url。

联系人和商务数据结构如 Contacts 与 Merchant 表所示，其中 contactId 对应一个具体用户，其余为该联系人基本信息。merchantId 对应一个具体商务，merchantLevel 为该商务用户等级，不同等级有着不同审核权限及区域，一个用户对应一个商务。

资讯详情数据结构如 News 表所示，newsId 唯一标识一条资讯，newType 用于区分资讯是图片资讯还是视频资讯，url 为视频链接，images 为新闻图集所对应的 url 集。

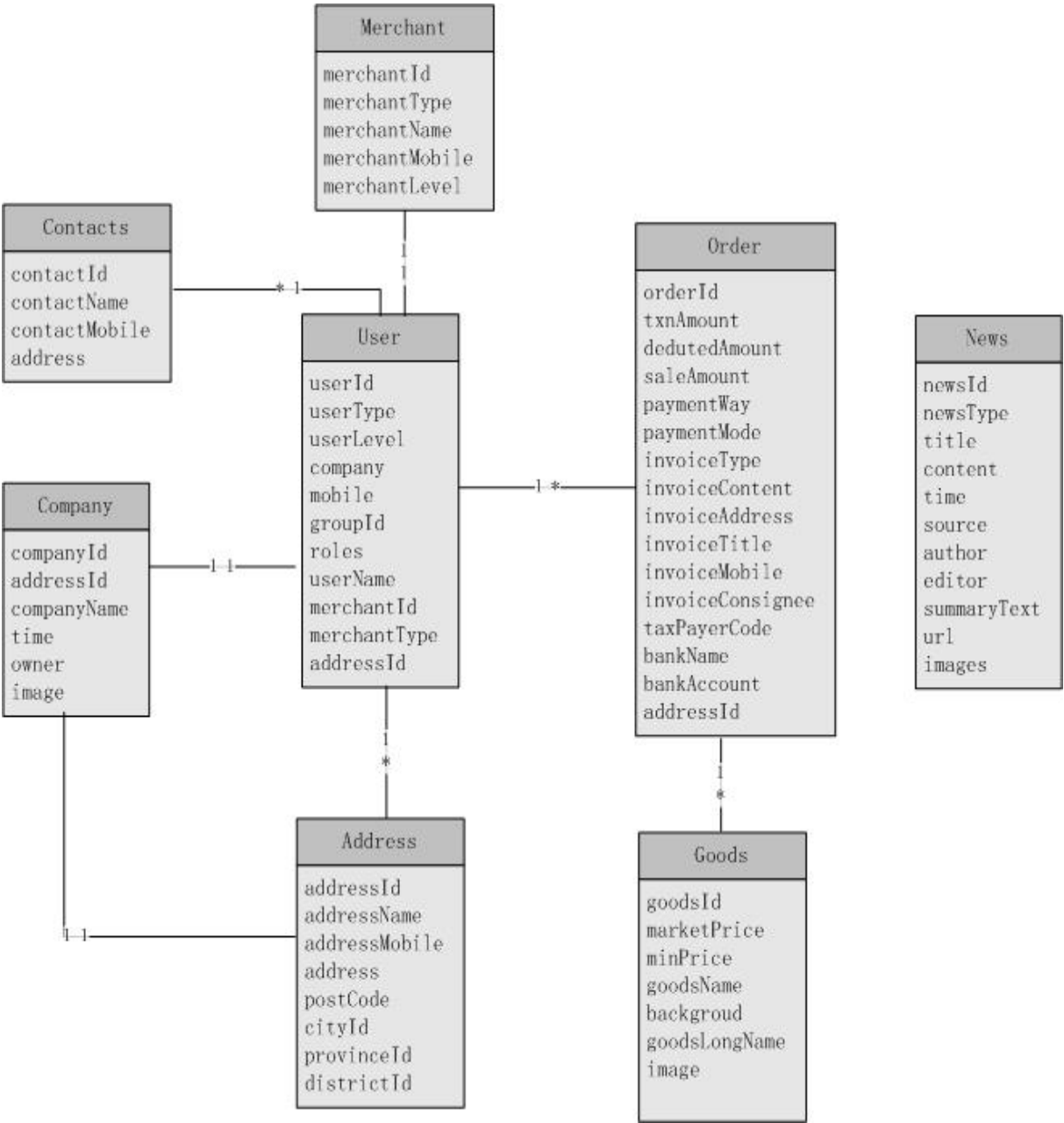


图 4-3 数据表结构图

4.4 本章小结

本章介绍了该客户端的整体架构设计，并对每层进行了详细的讲解，同时通过图解的方式梳理了各个模块之间的关系。接下来对客户端的每个功能模块的划分进行了简要描述，包括每个功能模块及其下的重要功能点。最后对客户端需要的主要数据结构进行了讲解，通过数据表结构图的形式展示了用户相关的表之间的关系，同时进行了相应的解释说明。

5 系统详细设计与实现

本章将根据功能需求部分划分的功能模块对该系统进行详细设计与实现，主要包含首页展示、票房分析、正品采购、行业资讯、商务订单以及个人中心六大模块。针对每个模块，通过流程图、类图并结合关键代码来具体描述。利用类图可视化各模块中的类和接口间关系，通过流程图详细描述模块中的业务流程。

5.1 首页导航的设计与实现

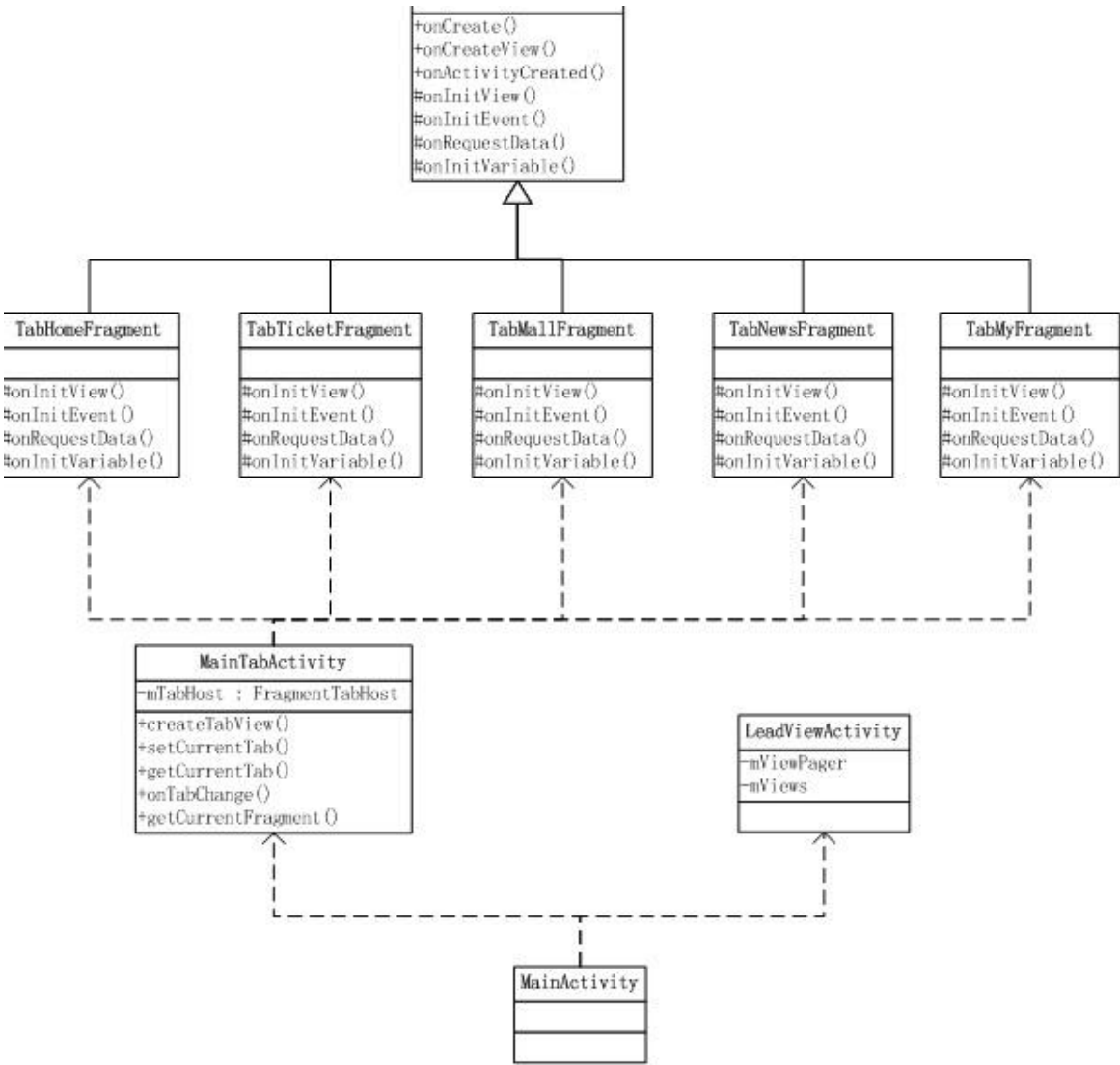


图 5-1 首页导航类图

首页导航的类图如图 5-1 所示。用户进入本系统时首先会进入 MainActivity 开屏页，MainActivity 会执行一个耗时两秒的 TimerTask 显示一张全屏图片，这张图片为本系统的宣传 Logo。同时根据存到 SharedPreferences 的 Int 变量判断用户是否在本机首次登陆，如果首次登陆，则进入 LeadViewActivity 引导页面。LeadViewActivity 通过一个包含五张全屏图片的 ViewPager 让用户迅速了解本系统的核心功能，用户滑动完毕后进入首页。如果判断本机已经登陆过本系统，则再根据后台配置决定是否展示广告，广告是一个三秒的全屏图片，点击后跳转至广告相关内容页，同时也可点击右上角关闭按钮进行关闭，广告展示完毕后进入首页。

进入首页后本系统采用底部 Tab 导航模式进行导航。底部 Tab 采用了将导航主体放置到首页底部，这样可以使得用户在使用过程中随意切换不同的页面。同时点击过的页面将会被缓存到内存中，再次切换到时无需重新加载，节省用户时间。

首页底部 Tab 导航包含五个 Tab 选项，分别为 TabHomeFragment、TabTicketFragment、TabMallFragment、TabNewsFragment、TabMyFragment，分别对应首页展示、票房分析、正品采购、行业资讯、订单管理五大模块。这五个 Fragment 均继承自 BaseFragment。BaseFragment 除了继承了 onCreate() 等基本方法，还有 onInitEvent()、onInitView()、onInitVariable()、onRequestData() 四个抽象方法，分别为初始化 CallBack、Listener 等事件方法，初始化 view，初始化私有变量、发出网络请求，四个方法均是在 Fragment onCreateView() 方法内进行调用。TabFragmentActivity 用于管理以上五个 Fragment，通过在 onCreate() 方法中获取到 mTabHost 实例，通过 addTab() 方法将五个 fragment 添加至其中，重新 onTabChanged() 方法进行相应的 Tab 选中逻辑处理。

5.2 首页展示模块的设计与实现

本节将具体介绍首页展示模块的各个子功能模块的详细设计与实现。包括广告展示、票房展示、商品展示、资讯展示模块的详细设计的介绍。首页展示的类图如图 5-2 所示。TabHomeFragment 即为首页，从上至下依次为用于展示广告的广告轮播图 BannerView，用于展示票房的自定义 View HomeTicketView，用于展示商品的自定义 View HomeMallView，用于展示自定义 View HomeNewsView。顶部 Titlebar 包含用于跳转扫描商品二维码 CaptureActivity 以及跳转购物车 CartActivity 的 ImageView。下面将针对各个功能点进行详细描述。

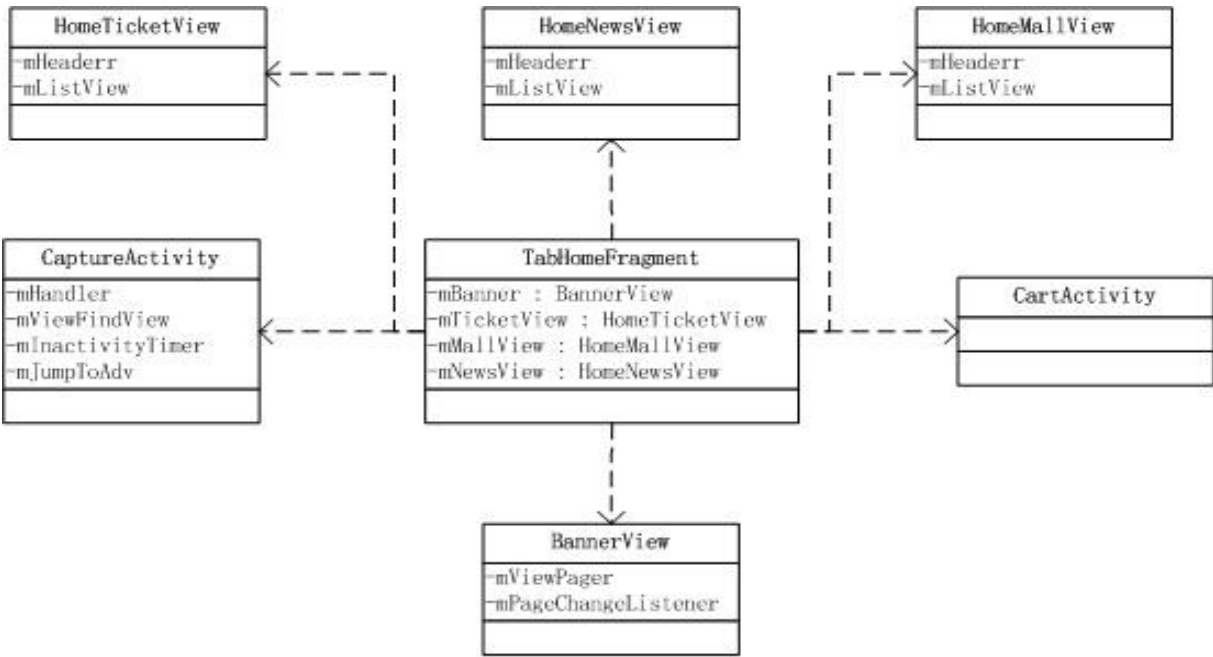


图 5-2 首页展示类图

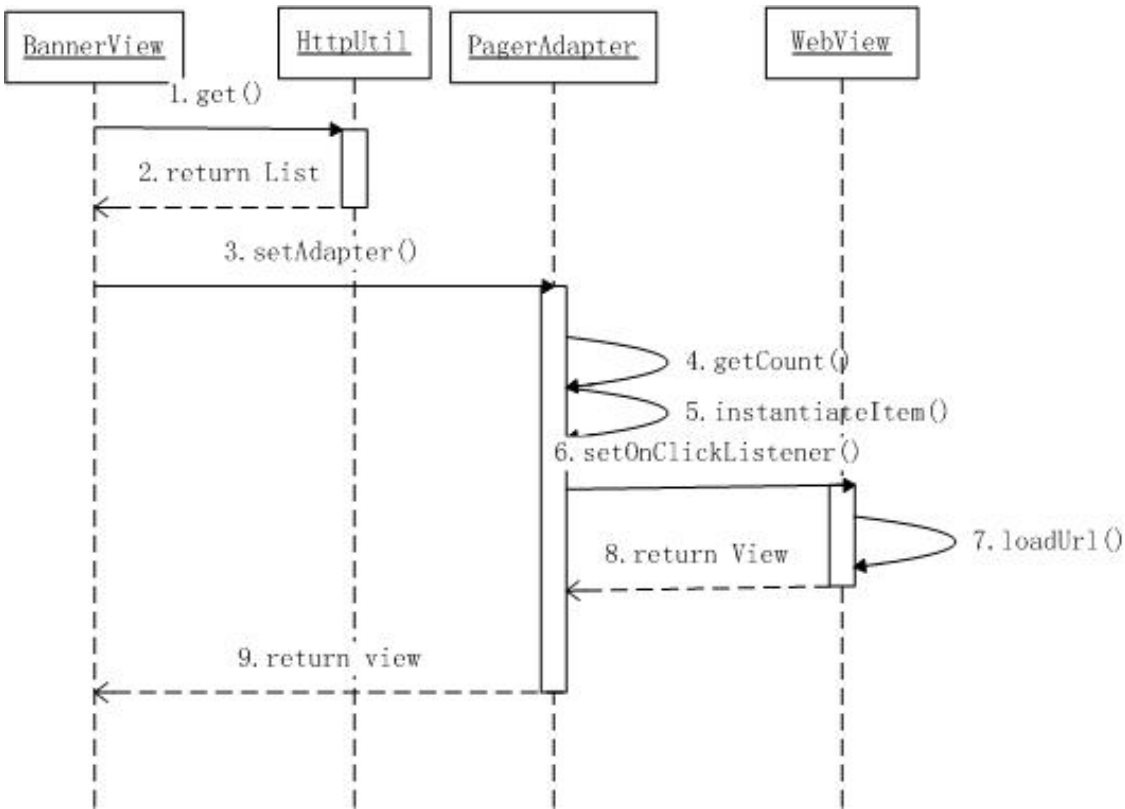


图 5-3 首页 BannerView 时序图

5.2.1 BannerView

BannerView 是首页顶部一个展示广告的轮播图，通过接口通过要展示的广告，其轮播效果通过 ViewPager 进行实现，BannerView 即为一个 ViewPager，其时序图如图 5-3 所示，主要实现步骤如下：

- 1) BannerView 通过调用 onRequestData()方法进行 HTTPUtil.get()请求接口，接口获取一个 List<HomeAdBean>数据集，每个 item 包含一个轮播图所需的数据 HomeAdBean，其主要数据如表 5-1 所示，然后调用 new ViewPagerAdapter(List<HomeAdBean>)创建 Adapter 对象；
- 2) 接着将调用 ViewPager.setAdapter(ViewPagerAdapter)方法将 Adapter 与 ViewPager 进行适配，以便能正确显示布局。ViewPagerAdapterAdapter 继承自 PagerAdapter，重写 getCount()方法，调用 List.size()获取集合大小并返回，ViewPager 通过返回值决定轮播图数量。同时重写 instantiateItem(ViewGroup container, final int position)方法，根据每个 HomeAdBean 去控制各个控件内容以显示布局；
- 3) 创建一个 Timer 对象，调用 timer.schedule(TimeTask)方法进行每隔三秒一次的广告切换，TimeTask 内部将通过 handler 通知 ViewPager 进行视图切换，重写 handler 的 handleMessage()方法，其中调用 ViewPager.setCurrentItem()方法，将视图切换到下一个。同时调用 mPager.setOnTouchListener()设置触摸监听函数，用户一旦触摸视图，则调用 Timer().cancel()方法取消 timer()，使轮播图不再自动化的，当用户手势一旦离开视图，则重新创建 Timer 对象进行自动滑动；
- 4) 最后调用通过 adapte.setOnClickListene()方法为 page 设置点击跳转，跳转页面通过 WebView 进行加载，其中 url 即为 WebView 需要加载的广告地址，调用 WebView.loadUrl()方法进行加载。

表 5-1 HomeAdBean 类

序号	属性	含义
1	gotoType	广告类型
2	url	图片地址
3	parameters	跳转及显示文字的参数

5.2.2 HomeTicketView

HomeTicketView 为一个继承自 ListView 的自定义 View，时序图如图 5-4，其主要

实现步骤如下：

- 1) 首先通过调用 `onRequestData()`方法进行 `HttpUtil.get()`请求接口，获取数据包括展示电影的 `List` 数据集和当日票房等数据。`List` 数据集用于填充 `ListView` 列表，通过调用 `mAdapter(List<HomeTicketBean>)` 创建 `Adapter` 对象，然后调用 `ListView.setAdapter()`方法将 `Adapter` 于 `ListView` 关联。`Adapter` 通过 `getCount()` 返回所需展示列表数量，通过 `getView()`方法填充每个位置的列表布局；
- 2) 当日票房等数据用于初始化 `HeaderView`。通过调用 `HeaderView.setContentView()`方法并将 `layout_home_ticket` 布局动态引入完成 `HeaderView` 初始化，调用 `header.init(HeaderData)`完成数据填充，通过调用 `ListView.addHeaderView()`完成 `HeaderView` 的动态添加；
- 3) 最后再为 `ListView` 动态添加 `footerView`，`footer` 即为一个 `TextView`，通过调用 `setOnClickListener()`方法并为其设置点击监听，当用户点击后跳转至票房分析模块。

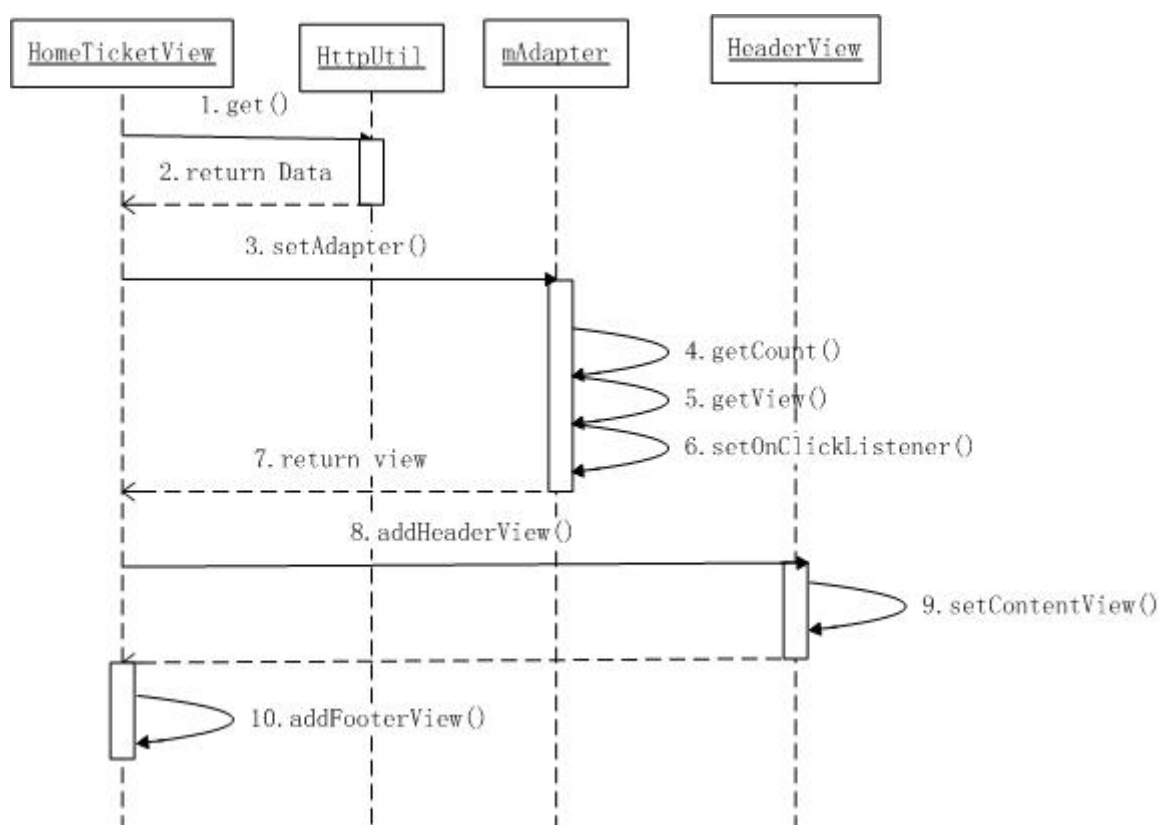


图 5-4 HomeTicketView 时序图

5.2.3 HomeMallView

`HomeTicketView` 是一个固定包含两个电影主题商品的自定义 `View`，其中每个电影

主题包含一个 header 和四个相关商品位, header 布局包含一个 `ImageView` 显示电影主题图片, 两个 `TextView` 分别显示电影名称, 商品数量等基础数据, 同时还包含一个 `Button`, 通过调用 `Button.setOnClickListener()` 为其设置点击监听函数, 点击后跳转自电影主题商品页。最后包含四个大小均等的 `ImageView` 进行商品展示。除此之外, 该 `View` 顶部为一个 `LinearLayout` 布局, 包含一个 `TextView` 和一个 `ImageView`, 点击后跳转自商品列表页。

5.2.4 HomeNewsView

`HomeNewsView` 为一个继承自 `ListView` 的自定义 `View`, 效果图如图 5-5 所示, 其主要步骤如下:

- 1) 首先通过通过 `layout_home_news` 布局动态引入并赋值给 `header`, 根据接口获取数据对其中控件进行数据填充并设置相应监听函数;
- 2) 将 `header` 动态设置为 `ListView` 的 `header`, 根据接口数据初始化 `Adapter`, `Adapter` 根据 `News` 的类型采用返回相应的布局模板, 每个 `item` 包含一个 `ImageView` 现实新闻图片, 三个 `TextView` 分别用于现实标签, 新闻标题和新闻梗概, 同时按照返回数据决定显示条数, 默认为三条;
- 3) 最后再为 `ListView` 添加 `footer`, 并为其设置 `onClickListener()`, 当用户点击后跳转至票房分析模块的其中一个子模块, 具体模块由接口数据决定。



图 5-5 HomeNewsView 效果图

5.2.5 CaptureActivity

CaptureActivity 用于商品二维码扫描，通过 Google 提供的第三方开源框架 zxing 进行设计，其主要实现步骤如下：

- 1) 首先创建一个 SurfaceView 对象，并通过 surfaceView.getHolder() 获取 viewHolder 将其传递给 initCamera() 方法进行相机初始化；
- 2) 同时实现 SurfaceHolder.Callback() 函数，在 surfaceCreated() 函数中再次调用 initCamera() 方法；
- 3) initCamera() 方法创建 CaptureActivityHandler 对象，代码如下：

```
private void initCamera(SurfaceHolder surfaceHolder) {  
    CameraManager.get().openDriver(surfaceHolder);  
    if (handler == null)  
        handler = new CaptureActivityHandler(this, decodeFormats, characterSet);  
}
```

- 4) handler 扫描到二维码后会调用 CaptureActivity.handleDecod() 方法并将跳转参数传递过来，该方法会通过传递的参数进行网络请求，从而跳转至商品详情页。

5.3 票房分析模块的设计与实现

本节将具体介绍票房分析模块的各个子功能模块的详细设计与实现。票房分析模块基于专业及用户的海量数据进行分析，全面分析电影排片和票房数据，涵盖影片、影院、院线的实时/历史票房查询、实时排片多维度的对比分析功能。票房分析模块的类图如图 5-6。

票房分析的入口为 TabTicketFragment，票房分析、排片分析、电影首发、电影比较、电影 / 院线 / 影院 搜索 分别对应 BoxOfficeActivity、CinemaTicketActivity、MovieReleaseActivity、TicketCompareActivity、SearchActivity。除 SearchActivity 外其它四个 Activity 均通过 WebView 打开，WebView 统一使用 BaseWebView，便于管理与修改，此外 TicketCompareActivity 的入口在票房分析内。

BaseWebView 类主要包含 urlType、scrollChangeAction、listener 三个属性，urlType 用于标识 WebView 请求的类型，不同类型采用不同模板显示，scrollChangeAction 是一个 Action 接口的实现对象，listener 是 WebView 的事件回调函数，比如返回等操作。四个通过 WebView 显示的 Activity 还包含一个原生实现的 TitleBar，即类图中的 title，不同的 Activity 展示不同的 title。其中 BoxOfficeActivity 的 title 还包含了一个 ImageView，

通过点击 `ImageView` 可以进入 `TicketCompareActivity` 进行票房对比。票房对比支持影院/院线及城市的多维度对比，其具体效果如图 5-7 所示。

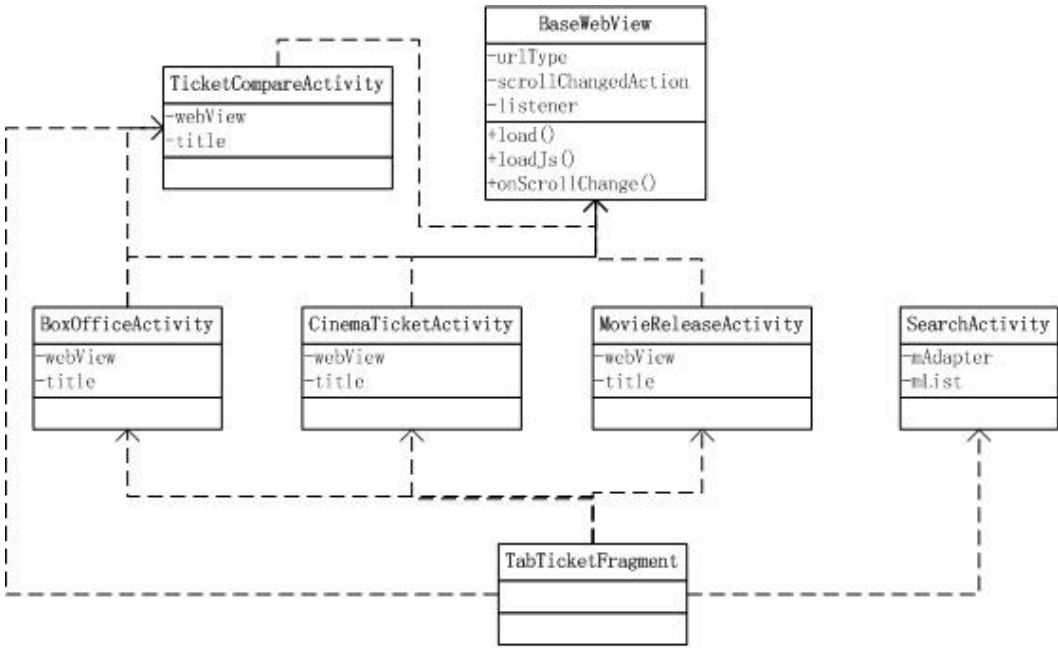


图 5-6 票房分析类图



图 5-7 票房对比效果图

除此之外票房分析的 title 也包含了一个 `ImageView`，点击进入 `SearchActivity` 页进行搜索。`SearchActivity` 包含一个自定义 `SearchView`，对输入进行监听，有新的输入便访问接口，根据接口返回的 List 数据集去初始化搜索建议 `mAdapter`，同时调用 `mList.setAdapter` 方法进行显示。下一次输入更改时直接调用 `mAdapter.setData()` 方法修改数据集，同时调用 `adapter.notifyDataSetChanged()` 方法改变显示的 `ListView`。如果无搜索建议则提示用户没有相关结果，请检查后重新输入。点击搜索下拉框数据可以进入相应的电影或影院详情。`SearchActivity` 还会通过一个自定义 `View` 实现推荐最新影片、最热院线等，同时支持按影院和影片分类进行筛选。

5.4 正品采购模块的设计与实现

本节将具体介绍正品采购模块的各个子功能模块的详细设计与实现。主要包括商城首页、商品列表、商品搜索、商品详情、购物车以及支付功能。正品采购模块的类图如图 5-8。

`TabMallFragment` 即为正品采购模块入口页，顶部采用一个用于展示热门商品的轮播图 `BannerView`，接着采用了用于展示热门商品和主题商品的自定义 `View` `HomeMallView`，顶部 `titlebar` 有一个用于搜索商品的 `EditText`，其侧边为一个放大镜样式的 `ImageView`，通过调用 `ImageView.setOnClickListener()` 设置监听事件，点击该 `ImageView` 后将通过 `getEditText()` 方法获取用户搜索内容，然后将其当做参数调用 `HttpUtil.get()` 方法进行网络请求，如果返回值为 `null` 则代表无搜索匹配项，弹出 `toast` 提醒用户，否则将会跳转至相应的 `MallListActivity` 页。通过首页更多商品按钮也可进入 `MallListActivity` 页，该页为商品列表页，用于展示所有商品，支持排序与筛选。点击具体的商品进入商品详情页，商品详情页主要通过 `WebView` 进行展示，可将其添加进购物车，通过点击 `titlebar` 上的购物车按钮可进入购物车页 `CartActivity`。`CartActivity` 可以对购物车内 `sku` 进行编辑，通过点击提交订单跳转至订单页 `PayActivity`。下面将针对各个功能点进行详细描述。

5.4.1 TabMallFragment

`TabMallFragment` 即为正品采购入口页，用于热门商品展示，精选商品推荐，让用户在商城首页便能看到购买频率最高的商品，同时提供了商品列表，商品搜索等入口。主要通过两个模块进行实现，其时序图如图 5-9 所示，主要步骤如下：

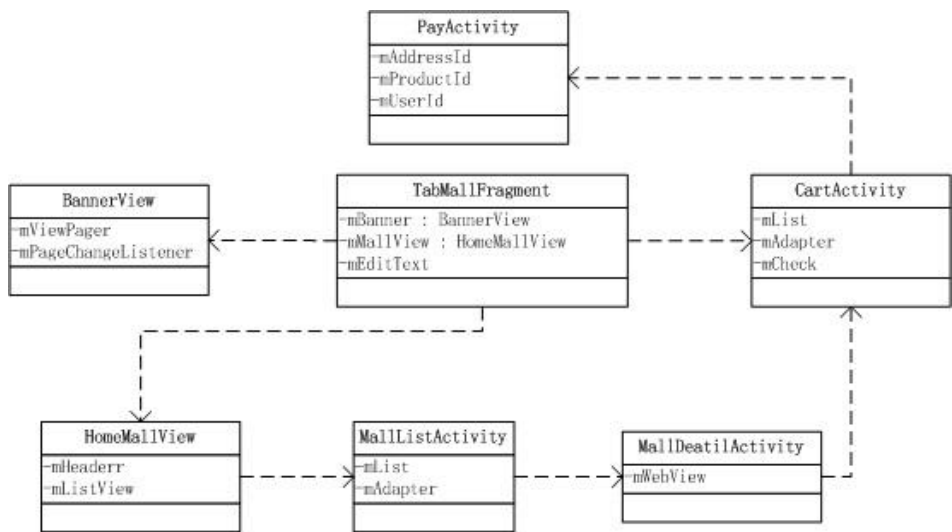


图 5-8 正品采购类图

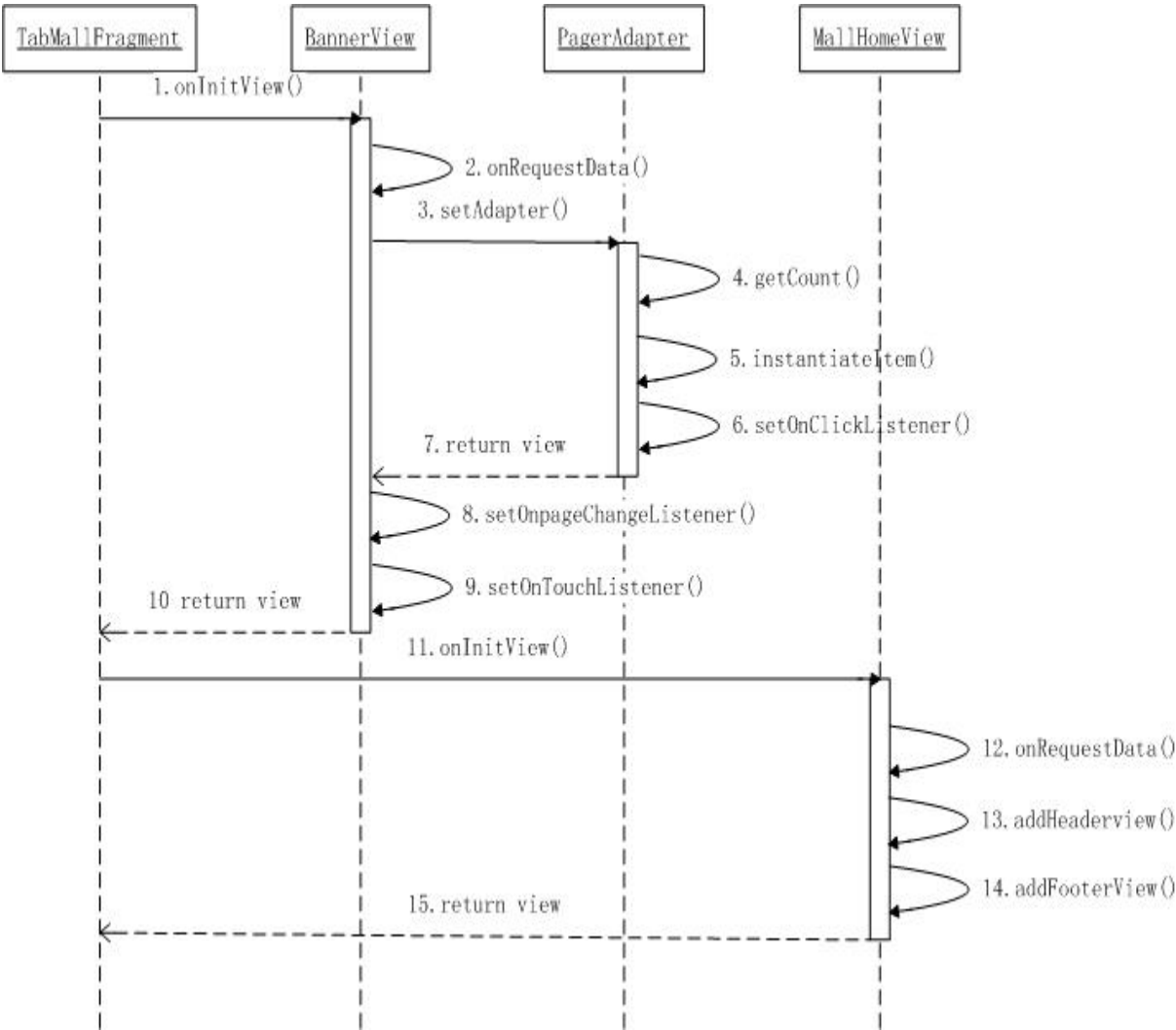


图 5-9 TabMallFragment 时序图

- 1) TabMallFragment 通过在 onInitview()方法内依次对两个 View 进行初始化显示。顶部为一个 BannerView，是一个用于进行热门商品展示的轮播图，BannerView 采用 ViewPager 实现，通过 onRequestData()方法访问接口，获取一个 List 数据集，每个 item 包含一个轮播图所需的数据 MallAdBean，通过创建获得数据创建 ViewPagerAdater 对象，然后调用 ViewPager.setAdapter(ViewPagerAdapter)方法将 Adapter 与 BannerView 进行适配。ViewPagerAdapterAdapter 继承自 PagerAdapter，重写 getCount()方法，调用 List.size()获取集合大小并返回，ViewPager 通过返回值决定轮播图数量。同时重写 instantiateItem(ViewGroup container, final int position)方法，根据每个 MallAdBean 去控制各个控件内容以显示布局，其中通过 position=position%getListSize() 实现轮播图循环显示效果，最后调用 container.addView()将 item 布局加入；
- 2) 完成 Adapter 初始化后将对其添加自动滑动效果，实现一个 OnPageChangeListener，重写 onPageSelected(final int position)方法，通过调用 handle.postDelayed(Runnable, time)并设置 time 为三秒，因此每隔三秒就会调用一次 Runnable 内的 run()方法，run()方法内将调用 Adapter.setId 设置当前应该显示的 item，同时调用 notifyDataSetChanged()完成界面刷新。同时调用 setOnTouchListener()设置触摸监听函数，用户一旦触摸视图，则重置定时器；
- 3) 接下来是两个自定义 View HomeMallView, HomeMallView 是一个 ListView 对象，通过 onRequestData()方法访问接口，获取 List 数据集，接着调用 ListView.setAdapter()进行 HomeMallView 显示初始换。之后再调用 HomeMallView.addFooter()和 addHeader()方法动态添加 HeaderView 和 FooterView；
- 4) 同时，HomeMallView 支持两种类型数据集，分别用于展示电影主题 ListView 和热门商品 ListView，通过接口返回的数据 data.getType()进行判断，Type 为 1 以热门商品 ListView 进行显示，Type 为 2 则以电影主题 ListView 进行显示，同时调用 HeaderView.setOnclickListener()和 FooterView.setOnclickListener，根据不同 type 值进行不同的点击跳转。

5.4.2 商城列表

商城列表页即为 MallListActivity，其主要步骤如下：

- 1) 商品主体通过 ScrollView+GridView 进行展示，根据不同尺寸手机分为每行显示三个或四个商品，每个商品 item 包含一张商品展示 ImageView 及显示商品名称的 TextView；

- 2) 商城列表提供两种选项进行商品分类，分别为商品类型和价格区间，同一商品类型例如居家生活的商品在同一个 GridView 内，每个商品类型名称通过一个 HeaderView 进行展示，同时与其他类型的商品分隔开，价格区间同理，每次进行选项选择后重新从接口获取数据；
- 3) 商城列表还支持通过电影主题筛选电影，筛选结果同样包含在 MallListActivity 类，只是其商品内容全部为该电影的衍生品，选项规则适用于此，其访问接口 URL 为：baseUrl + "Mall/MallList.api?type={0}&movieId={1}"，type 传递 1,2 分别表示按商品类型、按价格区间进行展示，默认为 1，movieID 传递 0 表示所有商品，否则展示相应电影的衍生品，默认为 0；
- 4) 商城列表顶部 titlebar 包含一个用于搜索的 EditText 框，输入选择条件后重载 MallListActivity，其访问接口 URL 为：baseUrl + "Mall/MallList.api?type={0}&filter={1}"，type 传递 1,2 分别表示按商品类型、按价格区间进行展示，默认为 1，filter 传递筛选条件。

5.4.3 购物车

购物车页为 CartActivity，其主要步骤如下：

- 1) 购物车通过 ListView 进行显示，每个 sku 占据一个 item，包含 sku 的展示图片 ImageView，数量，价格，名称等 TextView，每次进入购物车后从接口获取数据显示，用户可以通过每个 sku 左边的 CheckBox 选择框进行选择或取消选择，针对每个 sku 还可以通过+-两个按钮进行数量操作；
- 2) 通过购物车右上角编辑按钮可以进入购物车编辑页面，可以进行全选或依次选择，选择后可以集体删除或添加至收藏夹，同时针对单个 sku 可以直接进行移除购物车选项，点击右上角完成后回到购物车页；
- 3) 购物车最下方始终会呈现一个浮动的 BottomView，这个 BottomView 继承自 View，有一个全选 CheckBox，同时提供一个 TextView 显示已选择商品的总价(不含运费)。最后通过一个结算按钮进入订单支付页，结算按钮同时也是一个自定义 View，会始终显示已选商品数量。

5.4.4 提交订单

提交订单页为 PayActivity，提交订单页主要包含一个地址选择框、选择商品 List、商品总价、配送方式及时间选择、发票选择及提交订单。其主要步骤如下：

- 1) 地址选择框是一个自定义 View MallSelectView，包含一个地址 ImageView，三

个 `TextView` 分别为收货人、收货地址及联系电话，通过 `MallSelectView.setOnClickListener()` 为该 `View` 添加点击监听，点击后跳转至地址详情页 `MallAddressActivity`。地址编辑页包含收货地址的增删改查，采用 `ListView` 的形式展示每个地址的简要信息。通过管理按钮可以对现有地址进行修改和删除，通过点击修改跳转自地址编辑页 `MallAddressDetailActivity`；

- 2) 地址编辑页包含地址的详细信息，地址修改及添加都是通过此页实现，`MallAddressDetailActivity` 根据 `type` 决定如何展示，1 表示地址修改，2 表示添加地址。修改添加地址时通过点击保存按钮进行提交，点击保存后将会调用 `updateAddress()` 方法，`updateAddress()` 方法将会获取用户已经填写的信息同时进行信息验证，不满足要求的地址将会对用户进行相关提示，验证成功后进行网络请求，订单提交接口为 `Consignee/UpdateConsigneeAddress.api`，添加地址成功后会得到一个 `addressId`，同理修改和删除地址将传递 `addressId`。此效果图如图 5-10 所示；
- 3) 通过点击 `MallAddressActivit` 页内的任一个即为选中改地址，立即执行 `setResult(PayActivity.RESULTCODE_SELADDRESS)` 方法，同时 `finish` 掉当前 `Activity`，`PayActivity` 的 `onActivityResult()` 方法将进行判断，如果选择新的地址，则重新获取数据刷新当前 `Activity`；
- 4) 订单商品列表通过 `ListView` 进行显示，`ListView` 每个 `sku` 为一个商品，因为最外层已用 `ScrollView` 进行嵌套，所以需要重写 `ListView` 的 `onMeasure` 方法，代码如下所示：

```
public void onMeasure(int widthMeasureSpec, int heightMeasureSpec) {  
    int expandSpec = MeasureSpec.makeMeasureSpec(Integer.MAX_VALUE >> 2,  
        MeasureSpec.AT_MOST);  
    super.onMeasure(widthMeasureSpec, expandSpec);  
}
```

- 5) 配送及送货时间和发票信息选择都通过 `CheckBox` 的方式进行选择，选择成功后跳转 `PayActivity` 重新刷新；
- 6) 底部始终浮动一个 `BottomView`，显示当前订单总价及商品数量，还提供一个订单提交 `Button`，提交成功后跳转至支付页，在线支付外接第三方支付接口，提供微信支付，支付宝支付及银联在线支付三种方式，对于部分用户还提供货到付款形式，支付失效为两小时，再次期间可以联系商务进行订单修改或取消订单。

图 5-10 地址编辑效果图

5.4.5 商品详情

商品详情页为 `MallDetailActivity`，因考虑到商品详情需随时修改，同时提供大量图片进行详情描述，因此采用 `WebView` 加载，`MallDetailActivity` 通过调用 `WebView.loadJS()` 方法加载商品详情。顶部有分析按钮进行分享，外接微信、微博和 QQ 的分享接口，分享框包含商品基本信息，点击后通过浏览器跳转至该商品详情页。最下方 `BottomView` 有加入购物车和立即购买两个 `Button`，分别添加到购物车和跳转到订单填写页。

5.5 行业资讯模块的设计与实现

本节将具体介绍行业资讯模块的各个子功能模块的详细设计与实现。主要包括资讯列表、资讯详情及资讯分享页。行业资讯模块的类图如图 5-11。

`TabNewsFragment` 即为行业资讯模块入口页，顶部有全部、快讯、声音、解读和产业圈五个分类选项，五个选项分别为五个 `ListView` 实现，首先在 `onCreate()` 方法中调用 `initView()` 方法对所用到的 `View` 进行初始化，包括五个 `PullToRefreshListView`、`ViewPager` 及 `PagerSlidingTabStrip` 等。`PullToRefreshListView` 继承自 `ListView`，是一个封装好且支持上拉加载和下拉刷新的 `ListView`，这里我们仅将其设置为支持上拉加载，每一次上拉刷新载入 20 条数据，通过调用 `listView.setOnRefreshListener()` 方法设置上拉加载逻辑。然后创建一个包含 `View` 的 `List<View>` `pageViews`，调用 `pageViews.add()` 方法将五个 `ListView` 加入其中，接着对 `ViewPager` 的 `Adapter` 进行初始化：`viewPager.setAdapter(new`

NewsViewPagerAdapter(pageViews,titles));其中 titles 是五个 ListView 对应的 Title 字符串全部、快讯、声音、解读和产业圈。

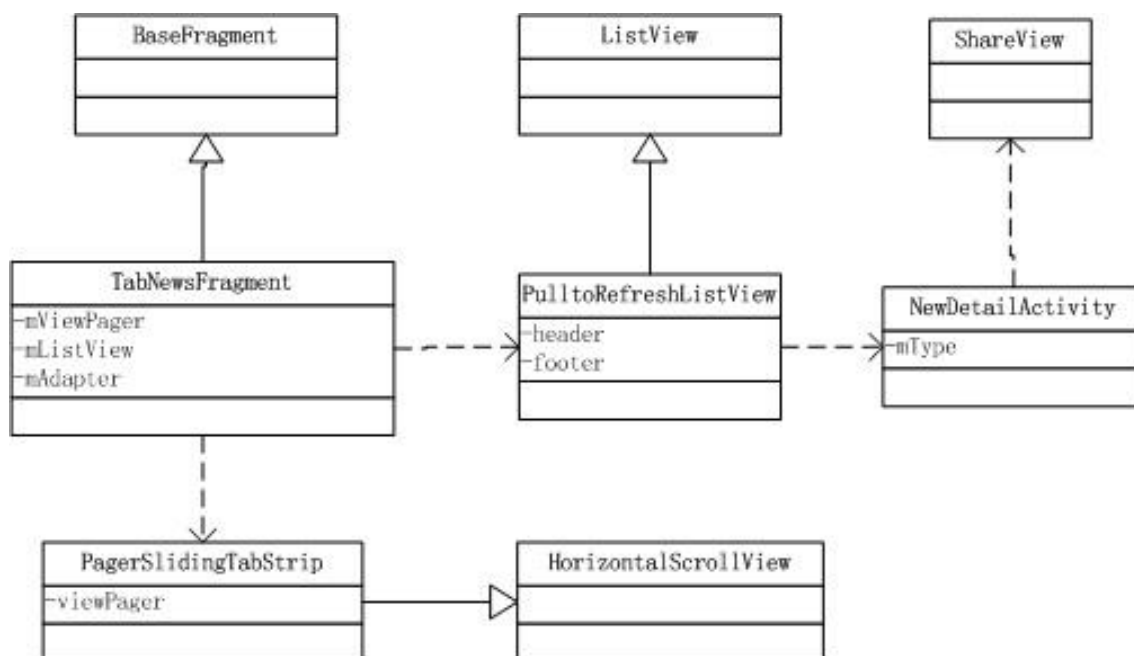


图 5-11 行业资讯类图

完成 ViewPager 初始化后 ViewPager 就可以进行后需将其添加至 PagerSlidingTabStrip, PagerSlidingTabStrip 是一个继承自 HorizontalScrollView 的控件, 调用 PagerSlidingTabStrip.setViewPager(viewPager)即可完成初始化。除此之外还需要为 tabs 设置监听页面切换函数, 其代码实现如下:

```

tabs.setOnPageChangeListener(new OnPageChangeListener() {
    @Override
    public void onPageSelected(final int position) { }
    @Override
    public void onPageScrolled(final int arg0, final float arg1, final int arg2) { }
    @Override
    public void onPageScrollStateChanged(final int arg0) {
        // 页面加载过来之后在请求数据, 不然在 onPageSelected 会出现卡顿现象
        if (arg0 == ViewPager.SCROLL_STATE_IDLE) {
            setSelector(viewPager.getCurrentItem());
        }
    }
});
  
```

其中 setSelector(viewPager.getCurrentItem())方法作用是根据当前页的标识请求数据, 如果数据已经初始化过了便不再进行数据请求, 其效果图如图 5-12 所示。



图 5-12 行业资讯类图

通过点击其中任一则资讯进入资讯详情页 `NewDetailActivity`, 资讯分为文字资讯、图片资讯和视频资讯, 文字资讯和图片资讯均采用约定好的 `Html` 格式进行显示, 所用的 `Html` 格式已经加载到代码内。视频资讯的视频播放通过一个继承自 `ViedoView` 的 `FullScreenVideoView`, 其增加功能为点击后自动全屏播放, 播放视频进度条通过 `SeekBar` 进行实现, 通过调用 `mSeekBar.setOnSeekBarChangeListener()` 为其设置监听函数, 监听函数通过对用户操作进行判断进行相关操作。顶部有分析按钮进行分享, 外接微信、微博和 QQ 的分享接口, 分享框包含资讯基本信息, 点击后通过浏览器跳转至该新闻详情页。

5.6 个人中心模块的设计与实现

本节将具体介绍个人中心模块的各个子功能模块的详细设计与实现。个人中心模块用于用户进行账户相关的信息查询管理, 主要包括查看订单、登录注册、账户管理、帮助中心等功能, 实现类图如图 5-13 所示。

`TabMyFragment` 即为个人中心模块入口页, 通过两个 `Button` 实现登录注册跳转, 接下来每一行通过一个自定义 `View` 提供其他模块的跳转入口, 自定义 `View` 从左到右包含一个模块 `Log` 图片, 模板名称和跳转制式箭头。

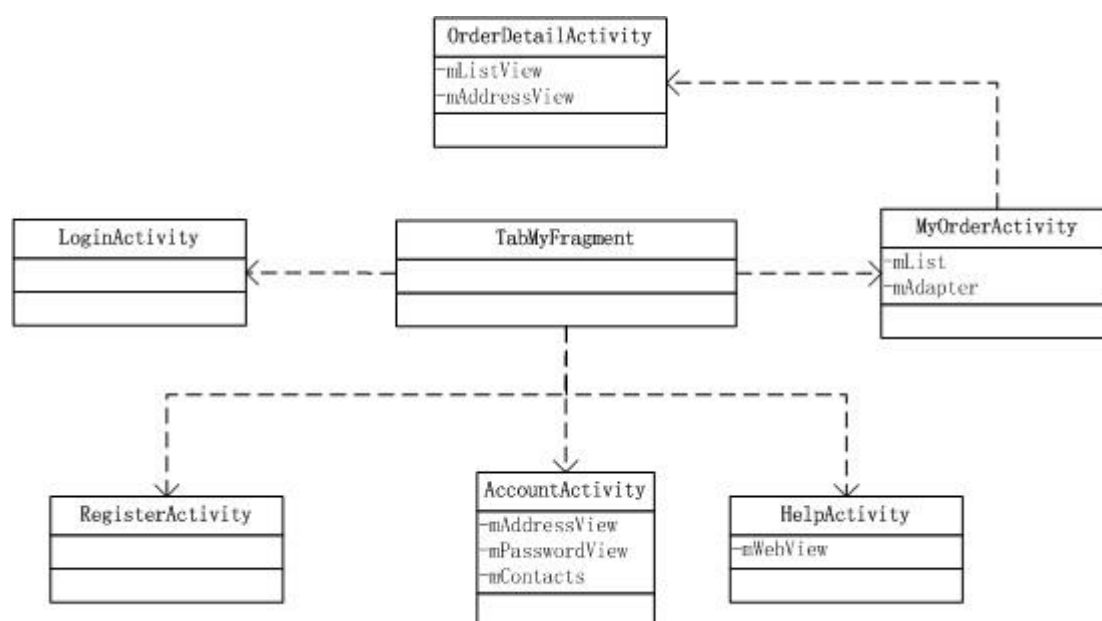


图 5-13 个人中心类图

登录功能采用两个自定义 **EditText** 框提供账号密码输入，账号即为用户手机号，连续三次登录失败会要求输入验证码，防止恶意用户进行恶意操作。用户点击登录后会访问 **Login.api** 进行登录，后台信息验证成功后返回用户信息。用户注册提供四个 **EditText** 进行信息输入，分别为手机号，密保及密码确认，验证码，点击注册后首先会在本地验证输入格式是否正确，若不正确则提醒用户重新输入，正确则访问 **register.api** 进行注册信息验证，后台验证成功后需补充填写公司及个人信息，个人信息将包含真实姓名和名片，真实姓名不可修改，信息补充成功后将等待后台人员审核，审核成功后方可进行用户操作。

MyOrderActivity 为我的订单页，我的订单页包含用户过往订单详情，支持全部、待支付和待收货三种分类，通过 **PullToRefreshListView** 实现，**PullToRefreshListView** 继承自 **ListView**，是一个封装好且支持上拉加载和下拉刷新的 **ListView**，这里我们仅将其设置为支持上拉加载，每一次上拉刷新载入 20 条数据，通过调用 **listView.setOnRefreshListener()** 方法设置上拉加载逻辑。每个 **item** 同时包含一个自定义 **ListView**，显示每个商品的基本信息，同时支持订单删除，订单删除通过访问后台接口进行实现，删除成功后将重新刷新当前 **Activity**，刷新后将不再看到该订单。通过 **refreshListView.setOnItemClickListener()** 为每个 **item** 设置点击监听，点击后跳转至订单详情页 **OrderDetailActivity**，**OrderDetail** 详情页包含订单的详细信息，这一页仅提供展示，不能进行其它操作，在此便不再赘述。用户进入我的订单页的流程图如图 5-14 所示。

AccountActivity 为我的账户页，分别包含联系人信息管理、收货地址管理及密码修改三功能，在此叙述收货地址管理页。收货地址管理页包含一个 **ListView**，每个 **item** 包

含三个 TextView 分别为收货人、收货地址及联系电话，同时两个 Button 分别为编辑和删除。通过 Button.setOnClickListener()为 Button 添加点击监听，点击编辑后跳转到地址编辑页 AddressDetailActivity, 包含地址的详细信息, 地址修改及添加都是通过此页实现, AddressDetailActivity 根据 type 决定如何展示, 1 表示地址修改, 2 表示添加地址。修改添加地址时通过点击保存按钮进行提交, 点击保存后将会调用 updateAddress()方法, updateAddress()方法将会获取用户已经填写的信息同时进行信息验证, 不满足要求的地址将会对用户进行相关提示, 验证成功后进行网络请求, 订单提交接口为 Consignee/UpdateConsigneeAddress.api, 添加地址成功后会得到一个 addressId, 同理修改和删除地址将传递 addressId。

HelpActivity 为帮助中心页, 通过一个 Titlebar 加上 WebView 进行实现, Titlebar 提供返回和标题展示, WebView 为帮助页主体, 包含大量文字图片描述, 每次新版迭代都会有所修改, 故采用 WebView 开发, 便于管理。

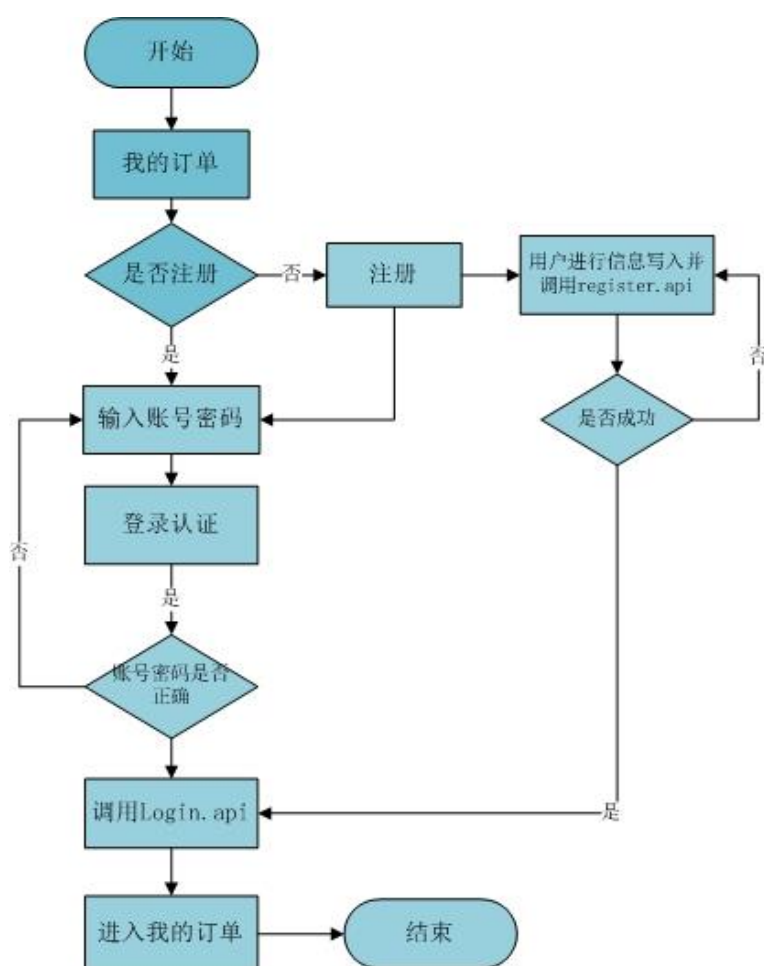


图 5-14 我的订单模块流程图

5.7 商务订单模块的设计与实现

本节将具体介绍商务订单模块的各个子功能模块的详细设计与实现。商务订单模块用于商务账号对负责订单进行查询管理，主要包括订单列表、订单详情、订单筛选等功能，实现类图如图 5-15 所示。

商务订单模块必须为商务账号才具有，商务账号为专门工作人员拥有，主要对负责区域的订单进行审核修改等操作，从个人中心模块我的订单页进入。进入页面即为订单列表页 `OrderListActivity`，`OrderListActivity` 通过列表的形式对订单进行展示，列表为 `PullToRefreshListView`，每页 20 条数据，下拉加载更多。每个 `ListView` item 包含订单的商品数量，订单总价等基本信息，还包含订单审核状态、订单号和用户所属公司名称。每个 item 下面有一个或两个 `Button`，分别为取消订单和查看订单详情，只有需要商务进行审核的订单显示取消订单按钮。订单列表页顶部有一个自定义 `View`，用与用户进行订单搜索，搜索支持公司名称、下单人、订单编号及商品名称四种方式进行搜索，点击搜索后访问接口 `baseUrl + "Order/OrderSearch.api?type={0}&search={1}"`；`type1,2,3,4` 分别对应公司名称、下单人、订单编号及商品名称，`search` 对应搜索输入，搜索字符应小于 30 个。搜索成功后将重新刷新当前 `Activity`。

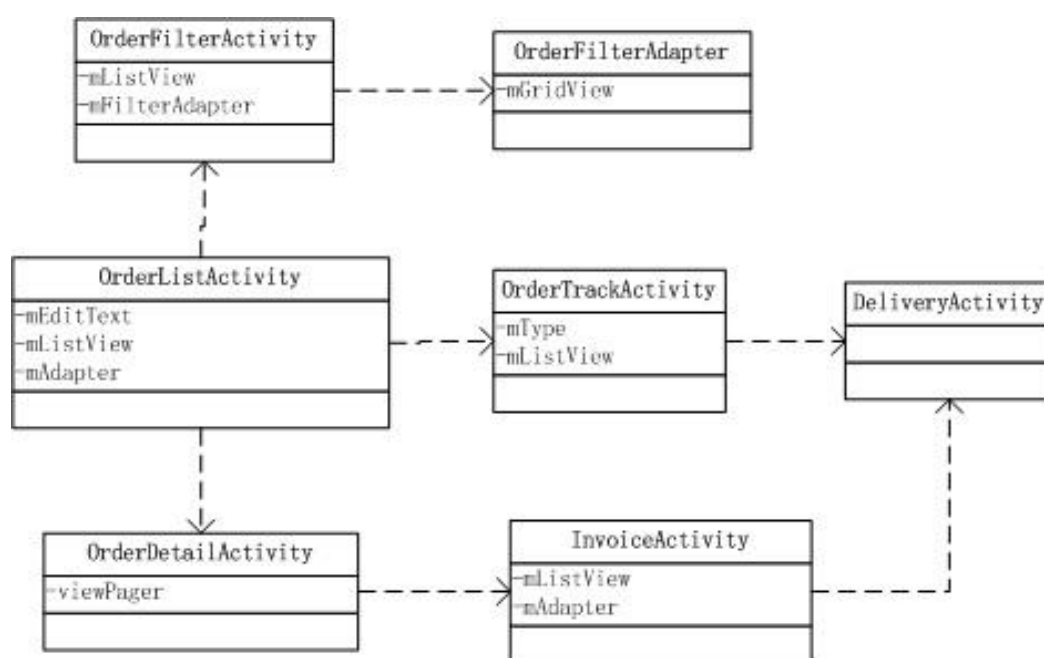


图 5-15 商务订单类图

通过在订单列表点击筛选可以进入订单筛选页，其时序图如图 5-16 所示。订单筛选页通过 `ListView+GridView` 的方式进行实现，包括区域名称和集团名称等筛选条件，每个筛选条件限选一个，每个筛选条件为一个 `GridView`，所有 `GridView` 包含在一个 `ListView` 中。这些筛选条件从服务器接口获取，接口会返回一个 `List<List<Object>>` 结构

的数据，然后使用数据集初始化 Adapter，Adapter 通过调用 `getCount()` 决定显示数量。同时调用 `getView(int position)` 方法进行布局初始化，每个数据集对应一个 `GridView`，`GridView` 在 `init()` 方法内使用数据进行布局填充，之后调用为每个 item 调用 `setOnClickListener()` 进行点击监听，点击后将 item 选中，同时将其数据记录到用于显示筛选条件的 Map 内部。当用户返回到订单列表页后将调用 `refresh()` 方法进行刷新，获取 Map 内的筛选记录，同时利用筛选条件筛选订单，将符合条件的订单显示到订单列表内。

点击订单列表页 item 的查看订单按钮将进入订单详情页 `OrderDetailActivity`，订单详情页包含订单 ID 和一个展示订单内商品的 `ListView`，因为 `ListView` 外层嵌套有 `ScrollView` 且每个订单商品类型不会太多，所以重写 `ListView` 将所有 item 显示出来，不支持滑动。同时订单详情页包含有收货信息，付款信息和发票详细信息。最下方将浮动三个按钮，分别为修改订单，审核通过和取消订单。`OrderDetailActivity` 分为两个 type，正常展示和修改显示，两种类型仅在显示上有区别，因此修改订单不跳转新的页面。修改订单时商品列表 `ListView` item 将显示一个用于修改价格的 `EditText`，同时发票选择有 `CheckBox` 进行相应选择，修改成功后跳转到订单列表页。审核通过按钮即对当前订单进行审核，点击后访问接口，成功后调用 `refresh()` 方法，同时对 `OrderListActivity` 进行刷新，删除订单操作相同。

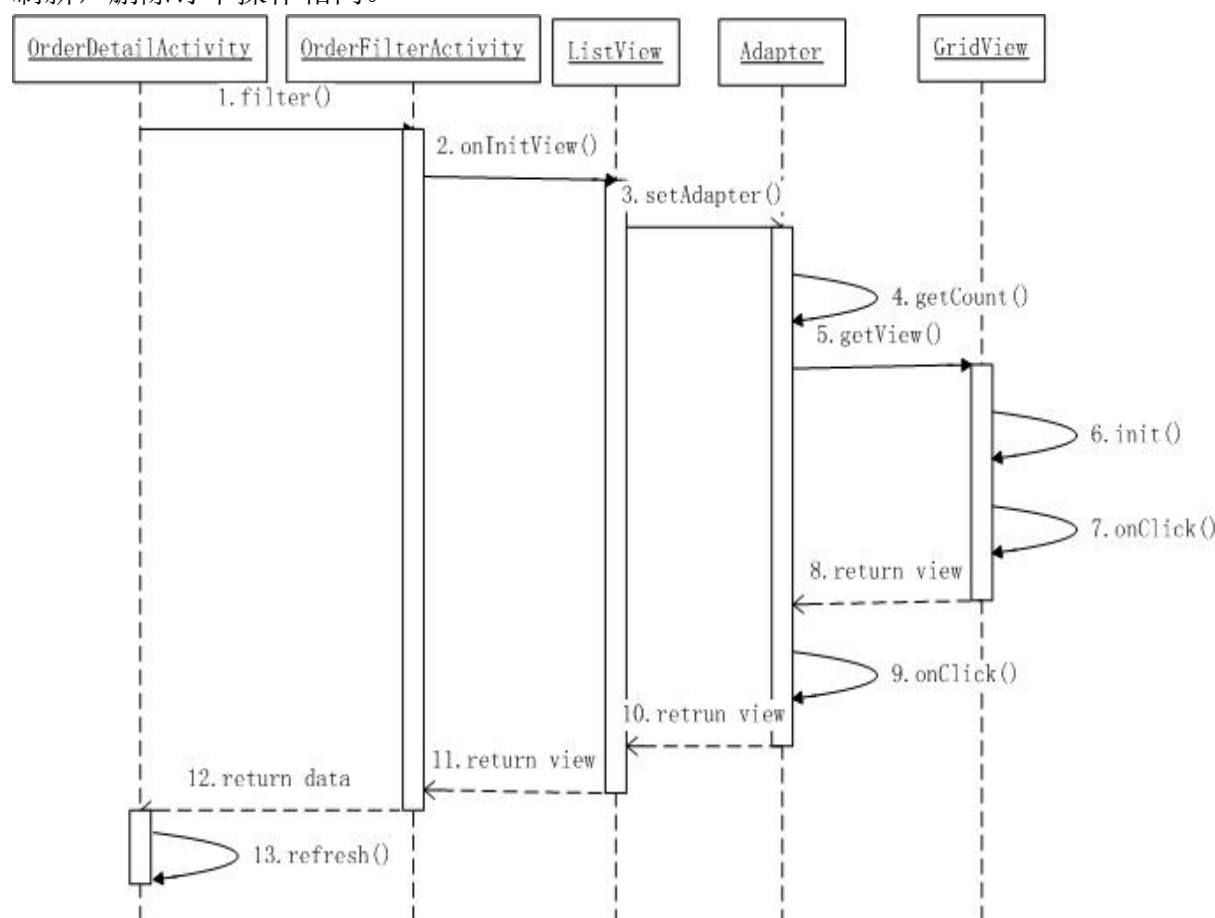


图 5-16 订单筛选页

订单详情页还可点击进入发票查询页 `InvoiceActivity`，每个商品对应一个发票，发票查询页通过原生 `ListView` 进行实现，不支持下拉刷新，每个 `item` 包含五个 `TextView`，分别显示发票号、发票金额、开票时间、发票类型和快递单号，还包含一个 `Button` 用于查看发票配送物流状态，点击后跳转自物流详情页 `DeliveryActivity`，物流详情页通过 `WebView` 显示，便于调用第三方快递接口及时更新配送状态。

订单列表内已发货的订单将会显示订单跟踪 `Button`，点击后跳转到订单跟踪页 `OrderTrackActivity`，顶部通过一个 `ListView` 显示订单的基本信息，包括订单编号、客户名称、收货人、收货地址和联系电话。发货详情将可能包含一个或多个 `ListView`，因为同一订单内的商品可能发货状态不一样，一起发货的商品在同一个 `ListView` 内进行显示，每个 `item` 显示商品的基本信息，同时通过一个 `Button` 支持物流查询，点击后跳转自物流详情页 `DeliveryActivity`，物流详情页通过 `WebView` 显示，便于调用第三方快递接口及时更新配送状态。同时对于未收货的商品提供确定收货 `Button`，点击和携带订单号和商品编号进行接口访问，访问成功修改收货状态。

5.8 本章小结

本章对该客户端进行了详细的设计，针对各个模块的功能给出了详细的设计与实现方案，通过模块设计的类图、流程图、功能截图等来描述模块的设计方法，并对关键代码进行了解释。

6 系统测试

为了保证系统的正确性，我们需要在特定的条件下，通过操作程序，从而发现系统可能存在的缺陷。测试工作是由公司内部的测试工程师完成的，本人也进行了部分测试辅助工作，主要是根据测试用例，同时编写部分自动化测试代码。

6.1 功能性测试用例设计

本节通过功能性测试用例，描述测试过程和结果，并针对测试结果进行相关说明。

表 6-1 添加收货地址测试用例表

用例编号	26587
测试优先级	高
用例标题	验证添加收货地址功能
测试类型	功能性测试
用例设计者	XXX
设计日期	2015 年 2 月 26 号
前置条件	用户已登录
测试方法	添加收货地址，观察操作结果是否正确
执行步骤	点击“收货地址管理”，点击底部“添加新地址”按钮，然后进行信息输入，对各个选项进行错误输入，例如手机号输入 8 位，不输入详细地址等，看是否能保存。之后输入完整的正确格式信息，点击“保存”按钮，通过抓包获取其 addressId，然后等待客户端跳转至收货地址列表，看刚才添加的地址是否显示，抓包观察其 addressId 是否与添加的地址一致，同时比对其余地址信息是否相同。
预期结果	输入错误数据时保存失败并提示用户，保存成功后地址列表显示正确且其信息跟自己填写的相同。
实际结果	与预期结果相一致。
结论	添加收货地址功能能正确运行。
对应开发人员	XXX
测试日期	2015 年 3 月 2 号

表 6-1 是对添加收货地址进行的测试。当用户进入收货地址管理页面并点击底部“添加新地址”按钮时将进行添加收货地址操作，收货地址包含收货人手机、姓名、省市及

详细地址等信息。本地会对输入数据进行判断，不合格的数据将给与提示，当数据验证合格后将访问接口，该用例主要作用为验证本地数据判断功能是否准确，同时在弱网情况下验证地址保存是否正确。

表 6-2 是对我的订单显示数量进行的验证测试。订单显示数量由接口返回，并不是根据返回 List 进行计算显示，因此验证接口返回显示是否正确是有必要的。

表 6-2 验证订单数量测试用例表

用例编号	26573
测试优先级	中
用例标题	验证订单数量显示功能
测试类型	功能性测试
用例设计者	XXX
设计日期	2015 年 2 月 26 号
前置条件	用户已登录且具有多个订单
测试方法	将订单加载完成，观察其数量是否与显示数量相同
执行步骤	点击“我的订单”，记录顶部显示的订单数量，滑动订单 ListView 并计算订单数量，待到无法上拉加载并提示已无更多订单时计算出订单总量，比对其与顶部显示数量是否相同，同时对多个账号进行测试。
预期结果	计算数量与顶部显示数量一致。
实际结果	与预期结果相一致。
结论	订单数量显示正确。
对应开发人员	XXX
测试日期	2015 年 3 月 2 号

6.2 性能测试用例设计

根据性能测试，能够验证软件系统的可靠性、稳定性、轻便性、易于维护、以及实用性。本节将通过一个性能测试用例，来验证系统的性能表现。

表 6-3 是首页展示模块的加载时间的验证。首页主要功能即为展示，因此首页包含了 4 个自定义 View，其中充满了大量的图片及相关数据，BannerView 所含的广告图片清晰度尤其高，因此首页即为客户端所有页面中加载数据最多的页，同时自定义 View 还需要大量渲染才能加载，因此也是加载时长最多的页。

因此验证系统响应性能通过测试首页即可，测试硬件为多部主流 Android 手机，手

机版本从 2.2 至 6.0，各版本比例按照 Google 所公布的 Android 市场机型分布比率进行调整，网络环境为移动通信联通 3g/4g 网络和公司 WiFi 网络，每 5 台手机为一组进行测试，防止网络等外界因素相互干扰影响测试结果。按照上述条件进行组合测试，每种机型每种组合测试 100 次，从经多次测试，首页加载时间约为 1.5 秒，其中网络传输为 1.3 秒，布局渲染为 0.2 秒，由此可见，加载大量数据的性能表现符合要求。

表 6-3 验证首页展示模块的加载时间

用例编号	27364
测试优先级	高
用例标题	首页展示模块的加载时间
测试类型	非功能性测试
用例设计者	XXX
设计日期	2015 年 3 月 2 号
前置条件	首页数据所在服务器运行正常，用户手机网络流畅
测试方法	打开客户端，加载首页数据，通过代码记录加载时间然后给出相应日志。
执行步骤	<ol style="list-style-type: none">1. 点击进入客户端；2. 等待广告加载完毕后开始记录时间；3. 等待客户端数据加载；4. 记录加载时间。
预期结果	首页加载时间在 2 秒内。
实际结果	首页平均加载时间约为 1.5 秒，其中 BannerView 加载仅需 0.3ms，HomeTicketView 等三个 View 需要 1.2 秒才能加载完毕。
结论	尽管首页图片及所展示数据较多，但是加载时间仍低于规定时间，用户可以流畅的加载客户端。
对应开发人员	XXX
测试日期	2015 年 3 月 2 号

6.3 本章小结

本章对比较有代表性的 2 个功能性测试用例展现了对系统功能测试的主要步骤，同时通过一个性能测试用例对系统性能测试进行了详细描述，包含其具体测试步骤及测试的软硬件环境等。通过本章进行的测试，有效保证了系统的基本功能性和性能。

7 总结与展望

目前我国电影产业正处于高速发展的阶段，影迷数量激增，越来越多的人已经养成了看电影的习惯。据统计，2015 年中国票房总量达到 440 亿元，仅次于美国排行世界第二，同时增速为 48.69%，远超美国，预计 2017 年中国将成为全球最大的票房产出国。同时中国衍生品行业资源严重缺乏，据了解，以美国为首的国外电影产业早已形成机制健全、信息充分流通的产业链生态圈，其行业利润不仅仅依靠票房，而是通过衍生品市场进入了新的高度。有数据显示，6 部《星球大战》全球票房收入 45 亿美元左右，但其衍生品收入已超过 200 亿美元。因此如何将电影产业链整合起来无疑成为现在的重中之重。

笔者主要的工作是参与 B2B 电影电商平台 Android 客户端的设计与开发。在此过程中，梳理了整个平台的业务流程，充分了解了客户需求，从最初的需求分析，再到概要设计、详细设计，再到系统开发，最终进行系统测试并上线，给客户带来了良好的用户体验，并且为其提供更加规范的一站式服务。

笔者在项目的整个开发过程中，应用了 Android 开发的知识，使用了多种大型开源工具及工具对系统进行开发。在实际项目中，笔者的系统设计能力和实现的能力得到了良好的锻炼。笔者在毕业论文中，通过对类图，对系统的实现做了详细的解读，通过界面截图，为读者带来了更直观的视觉感受。

目前笔者参与设计与实现的客户端已经在业内召开了产品发布会并正式上线，得到了业内人士的广泛好评，同时越来越多的业内人士开始下载使用，目前一切运行正常。

由于业务的扩展，该客户端还需要进一步完善才能满足用户需求，下面将主要介绍两点待完善的内容：

（1）目前系统并未实现用户信息行为采集，采用的是第三方 SDK 进行统计，无法对数据真实性进行校验，同时仅能记录一些用户点击的普通统计，不能针对系统的业务需求进行统计。因此未来将会开发一套配套的统计工具，能针对自身需求进行定制，例如统计用户从进入客户端到采购成功的路径等业务需求。

（2）由于时间有限，行业资讯页的资讯目前还不能支持评论功能，将在下一版本增加此功能。允许用户对资讯进行评论，对资讯评论列表进行展示，除此之外用户还能在资讯评论页回复其他用户的评论，此时还将添加评论通知系统，让用户能够及时收到别人的评论，同时还需在个人中心内增加我的回复及评论入口查看过往评论，支持跳转到该评论页。

参考文献

- [1] 龚成莹, 邢敬宏, 胡银保. 基于 JSON 的 Android 移动终端与 PHP 及 MySQL 数据通信[J]. 工业仪表与自动化装置, 2013(1):63-65.
- [2] 张欣. 基于 Android 的移动电子商务系统的设计与实现[J]. 电子世界, 2013(22):140-142.
- [3] 宋冬生. Git——版本管理之利器[J]. 程序员, 2007(11):118-119.
- [4] 赵艳. Android 中 bitmap 引起内存溢出的解决方案的对比分析[J]. 科技传播, 2013(24).
- [5] 申甲龙. 基于 Android 平台的新闻客户端的设计与实现[D]. 北京交通大学, 2014.
- [6] 梁永幸. 浅谈敏捷开发与其他传统开发方式的区别[J]. 电子世界, 2012(24):128-128.
- [7] 李难难. 移动电子商务终端的界面设计研究[D]. 上海交通大学, 2014.
- [8] 鲍晓. 基于 Android 平台的新闻资讯阅读软件的设计与实现[J]. 计算机应用, 2013, 33(z2).
- [9] 孟远. Android 网络通信框架 Volley 的解析和比较[J]. 软件, 2014(12):66-68.
- [10] Urbaczewski A, Valacich J, Jessup M L. Mobile commerce: Opportunities and challenges. Communications of the ACM, 2003, 46(12): 31-32.
- [11] 闵庆飞, 季绍波, 曲刚. Mobile Commerce User Acceptance Study in China: A Revised UTAUT Model[J]. Tsinghua Science & Technology, 2008, 13(3):257-264.

致 谢

首先，感谢我的毕业论文指导老师 XXX 老师，从实习、毕业设计开题、初稿完成直到最终定稿，XXX 老师给予了我非常大的帮助和支持。从论文的内容、论文编写的方法、规范等方面均对我进行了悉心的指导，帮助我开拓思路，谢谢老师。

其次，我要感谢在 XXX 实习时我的项目组同事。该项目为我在实习期间完成的项目，感谢我们组的各位同事，感谢他们对我在编程方法、编程规范、专业知识等方面进行的指导，我不仅在专业知识上有了提高，也学到了许多工作的经验。

还要感谢北京交通大学软件学院的各位老师对我在本科期间的教诲，感谢家人和朋友对我的支持和鼓励，他们的关心是我不断进取的动力。

最后还要感谢评阅笔者论文的各位老师，感谢你们认真细致、科学严谨的论文评阅工。

附 录

附录 A 参考文献翻译

The UTAUT model without revision cannot be applied to m-commerce user acceptance research since all IT adoption theories or models, including UTAUT, were developed for PC and/or fixed line Internet systems/applications. There are fundamental differences between m-commerce and Internet-based e-commerce in terms of network infrastructure, application development, client devices, and business model, e.g., value proposition, cost structure, and source of revenue. Internet-based e-commerce and mobile devices based m-commerce can be briefly explained as follows.

As pointed out by Heijden, different usage of IS leads to different adoption behavioral factors. For example, for hedonic IS adoption, perceived enjoyment/playfulness is more important than perceived usefulness. In fact, the majority of m-commerce applications today are for personal consumptions, that is, for personal leisure and play. Hence, UTAUT model's "performance expectancy" should be replaced by "utility expectancy", i.e., playfulness, satisfaction, leisure, and quality of life. Similarly, "voluntariness of use", one of UTAUT model's moderating factors, will be insignificant for individualized system. Furthermore, due to the newness of m-commerce, "experience" will be less influential in determining m-commerce user acceptance. Eliminating "experience" will also simplify the model. Of course, we can include these moderating factors in future studies.

One does not need to consider the cost of information systems if he or she uses the IS for work purpose because his/her employers will bear the cost. The cost of information systems based on the wired Internet is relatively low. In contrast, the cost associated with wireless transaction is still relatively high and many wireless users have to bear the cost themselves. Therefore, cost should be considered in m-commerce user acceptance. Wu and Wang and Fang et al. indicated that cost factor is negatively related with user's intent to use in m-commerce. We think that cost is user's assessment of the object world or reality, not his/her perception. Cost factor should thus be included in UTAUT's facilitating conditions. It will directly affect the users' adoption behavior. We believe that cost factor will significantly affect m-commerce user acceptance.

Consumer trust is recognized as one of the most important factors in the studies of both e-commerce and marketing. In other words, trust is an important positive attitude with respect to online purchasing behavior. Previous studies had introduced the concept of "perceived risk". In fact, perceived risk is closely related to trust. In traditional IT adoption theories, research focused on the acceptance behavior of technical

system itself. But trust is also reflected through their “trust” of the technology. Trust of technology is indirectly reflected on the belief of “utility”. In case of m-commerce, user acceptance is not only the acceptance of technology but also the acceptance of m-commerce service providers. For example, Siau and Shen classified trust into two categories: trust of technology and trust of m-commerce service providers. In addition, since m-commerce is still in its early development stage, many technical protocols, transaction standards, regulating policies, and payment systems need to be standardized. Therefore, trust is especially important for user acceptance. Earning user’s trust is critical for m-commerce’s success. It is necessary to introduce the concept of trust into m-commerce research, especially user’s trust of m-commerce’s service providers. Trust should be included in such concepts as usefulness and ease of use. There are rich literatures and theories on consumer’s trust of suppliers and service providers. Among these literatures, Bhattacharjee provided the definition and measurement of the consumer’s trust of an e-commerce service provider. The consumer’s trust of an e-commerce service provider is determined by three beliefs: ability, integrity, and benevolence. Ability refers to the user’s perception of provider’s “competencies and knowledge salient to the expected behavior.” Integrity refers to user’s perception that the service providers “will adhere to a set of principles or rules of exchange” acceptable to the users during and after the exchange. Benevolence refers to the extent to which a service provider “is believed to intend doing good to the users, beyond its own profit motive”.

One of the characteristics of mobile communication technology is the capability of identifying a mobile device user’s location. Many believe that mobile technology provides a great commercial potential for location-based applications and services. Although the capability may provide more personalized service, i.e., service providers can provide information services such as advertising and navigation based on the user’s location, it also poses potential privacy problems since the service providers will know the exact location of the user and might even know the user’s travel pattern. As a result, consumers are concerning with their privacy protection. There is a close relationship between privacy protection and trust. Bhattacharjee suggested that service providers should permit a user to choose how his or her personal information is used. This is an important measure of trust. However, privacy protection is different from trust because perception of privacy protection is also affected by the third party, such as government and its privacy protection policy. We think that as individuals become more aware of the importance of privacy, they will concern with the protection of their privacy. It is necessary to introduce privacy and trust issues in m-commerce user acceptance research.

Venkatesh et al. introduced gender and age as moderating variables in UTAUT model. They verified that demographics factors do moderate the relationship between effecting factors, the independent variables, and user acceptance, the dependent variable. Similarly, different types of users take different attitudes and acceptance behaviors toward m-commerce. We think that due to the unique characteristics of

m-commerce there will be greater differences in case of m-commerce than e-commerce. Okazaki indicated that demographics characteristics, such as gender, age, profession, income, marital status, and family structure, affect m-commerce's user acceptance in the case of Japan. Using cluster analysis, Okazaki found that young, affluent, and single women tend to be the most positive toward m-commerce. On the other hand, well-educated professionals tend to be the least positive. This may be one of the reasons that the development of m-commerce is slow. Hence, we should not neglect user's demographics when studying m-commerce user acceptance. We proposed to combine gender and age factors and form a single factor, the user demographics, and to include profession, income, and marital status in the revised model.

IT adoption research has been criticized for paying too much attention to perceived use behavior and not enough attention to user's satisfaction of the system itself. Therefore, the revised UTAUT can extend its scope by including user satisfaction. Research in user satisfaction in IS field is useful for us to understand IT adoption. According to user satisfaction theory, satisfaction is determined by a number of constructs. It consists of system quality, information quality, and related service quality. Wixom and Todd defined satisfaction as user's attitude toward information system itself, system quality, and information quality. These are object-based attitude and belief. In IT adoption, object-based attitude and belief are reflected by intent to use, usefulness, and easy of use. Specifically, it is believed that the higher the user's satisfaction toward the system, the higher the user's belief of system's ease of use. Hence, we can integrate the research findings in user satisfaction into IT adoption by introducing such system characteristics as system quality and information quality into the IT adoption framework. As shown by Wixom and Todd's empirical evidence, the inclusion of these system characteristics will increase our ability to predict a user's behavior in an IT adoption study.

Since there are significant differences in user interfaces, transmission speed, input and output, information richness, network compatibility between wired Internet-based e-commerce and m-commerce, m-commerce users tend to be more sensitive to these system features. Therefore, we believe that it is necessary to include mobile information system features in user acceptance research. System features consist of the user's attitude toward system satisfaction and information satisfaction. System satisfaction consists of reliability, timeliness, and integration. Information satisfaction is determined by currency, accuracy, format, and completeness. To this aspect, Ives et al., Melone, and DeLong and McLean had provided a solid theoretical foundation. However, further adjustments are needed for different types of m-commerce applications when applying these concepts.

Information technology is not culturally neutral. By the same token, the development of m-commerce will be most likely affected by a country's or region's social, economical, and cultural background. For example, NTT DoCoMo's tremendous success is believed to be unique and cannot be replicated by others. The unique Japanese culture is one of the reasons that explain the exploding penetration speed of iMode.

No IT adoption theory can explain such phenomenon. One possible explanation is the unique Japanese culture. It is not uncommon for researchers to include cultural factors in IT adoption research. The purposes of these studies were to explore the impact of culture on user's adoption decisions and to examine the applicability of IT adoption theories in different countries. For instance, Straub et al. found that TAM theory's validity must be significantly "discounted" for Japan compared with North America. In a similar study, Harris et al. found that although both UK and Hong Kong have the same wireless communication infrastructure, significant differences exist in m-commerce user's attitude and user behavior towards m-commerce. Evidence from many empirically based studies showed that cultural factors affect the degree of the importance of the determinants, the independent variables, on user's acceptance, the dependent variables, and moderate their relationships.

For research in m-commerce user acceptance in China, we think that culture should be considered as an important moderating factor, which should be added to the revised UTAUT model to make the model more applicable to the Chinese situation. To this end Hofstede, Martinsons and Westwood, and Srite and Karahanna provided us with a theoretical foundation for defining and measuring Chinese culture. In addition to such traditional cultural dimensions as power distance, uncertainty avoidance, individualism and collectivism, and masculinity and femininity, we might need to consider additional cultural constructs for m-commerce user acceptance research.

To study m-commerce user acceptance in China, we have proposed a revised UTAUT model that incorporates the factors of the unique characteristics of m-commerce, Chinese culture, and user satisfaction. We recommended that trust, privacy protection, and cost should be included in the revised model. We propose to introduce system specific moderating factors, namely, system quality and information quality.

After the theory establishment, the following must be done in the future to test the revised model: formulating proper research questions, designing a research that's appropriate for the nature of the research questions; developing a research instrument to measure the constructs of the research model, and collecting and analyzing data to validate the model. To empirically test our revised UTAUT model, multiple research methods including cross-sectional, longitudinal case studies, focus group approaches, etc. should be applied to understand user acceptance issues in different stages of m-commerce development as m-commerce evolves.

未经修改的 UTAUT 模型不能应用于移动商务用户接受度调查，因为所有信息技术采用的理论或模型，包括 UTAUT 模型，都是用于 PC 开发和/或固定线路网络系统/应用。在网络基础设施，应用开发，客户端设备和商业模式，例如，价值主张，成本结构和收入来源方面移动商务和基于互联网的电子商务之间存在着根本性的差异。基于互联网的电子商务和基于移动设备的移动电子商务可以如下简单地解释。

基正如 Heijden 指出，信息系统的不同使用将导致采用不同行为因素，例如，对于采用享乐 IS，

感知享受/嬉戏比感知有用的东西更重要。事实上，今天大多数的移动商务应用都用于个人消费，也就是个人休闲与娱乐。因此，UTAUT 模型的“性能预期”应该被“实用预期”所取代，例如娱乐、满意度、休闲和生活质量。同样，作为 UTAUT 模型的调节因素之一，“自愿使用”对个人系统来说将是微不足道的。此外，由于移动电子商务的新颖性，“体验”在决定移动电子商务用户接受度方面的影响将会减少。消除“体验”也将简化模型。当然，我们在今后的研究中也会包含这些调节因素。

如果为了工作而使用信息系统，你并不需要考虑信息系统的花费，因为你的雇主会承担这笔费用。基于有线因特网的信息系统成本相对较低。与此相反，无线交易相关的成本还是比较高的，许多无线用户需要自己承担这个成本。因此，在移动商务用户接受度中成本应该被考虑。吴、王和 Fang et al 表明，在用户考虑使用移动电子商务时，成本因素是负相关的。我们认为，成本是客观世界或现实用户的评价，而不是他的看法。成本因素应该因此被列入 UTAUT 模型的促进条件。这将直接影响用户的采纳行为。我们认为，成本因素将会显著影响移动商务用户的接受度。

在电子商务和市场营销研究中，消费的信賴都是最重要的因素之一。换言之，信任对于网络购买行为来说是重要且成正相关的。以前的研究曾提出“感知风险”的概念。事实上，感知风险与信任是息息相关的。在传统的 IT 采纳理论中，研究主要集中在技术系统本身的接受行为。但信任也通过自己的技术“信任”的体现。技术信托间接反映在“实用”理念上。在移动电子商务中，用户认可不仅是技术的认可，同时也是移动商务服务提供商的认可。例如，Siau 和 Shen 将信任分为两类：技术信任和移动商务提供商信任。此外，由于移动电子商务仍处于初期发展阶段，许多技术的协议，交易标准，规范的政策和支付系统需要标准化。因此，信任在用户接受度方面尤其重要。赢得用户的信任是移动电子商务成功的关键。有必要将信任的概念引入到移动商务研究中，特别是移动电子商务服务提供商的用户信任。信任应包括有用性和易用性的概念。在消费者的供应商和服务商的信任方面有着丰富的文献和理论，在这些文献中，Bhattacharjee 提供了用户对电子商务服务提供商信任的定义和测量标准。用户对电子商务服务提供商的信任被三个理念决定：能力，诚信，仁。能力是指用户对供应商“超出预期行为的能力和知识”的预期，诚信指用户对供应商在交易前后“会遵守交易准则”的预期，仁是指服务供应商“被认为在超出他们利润驱动的情况想会把好的给用户”的程度。

移动通信技术的一个特点就是有能力识别移动用户的位置。许多人认为，对于基于位置的应用和服务，移动技术有着巨大商业潜力。尽管这种能力可提供更个性化的服务，即服务提供商能够提供基于用户位置的广告和导航等信息服务，它也带来了潜在的隐私问题，因为服务提供者将知道用户的准确位置，甚至可能知道用户的出行模式，结果就是用户将会很关心他们的隐私保护。隐式保护和信任有着紧密的联系，Bhattacharjee 认为服务提供商应该允许用户选择如何使用他的个人信息，这是信任的重要措施。但是，隐式保护与信任又是不同的，因为隐私保护的准则会被第三方所影响，例如政府及其个人隐私保护法。我们认为，个人越来越意识到隐私的重要性，他们将关心他们的隐私保护。，有必要在移动电子商务用户接受度研究中介绍隐私和信任问题。

Venkatesh et al.介绍了 UTAUT 模型中的调节变量：性别和年龄，他们证实在独立变量影响因素

和因变量用户接受度之间，人口统计因素有中度影响，同样，对于移动商务，不同类型的用户将采取不同的态度和接受行为。我们认为，由于移动电子商务的独有特点，这种情况下移动电子商务相比电子商务会有较大的差异。Okazaki 研究了在日本人口统计特征对移动电子商务用户接受度的影响，如性别、年龄、职业、收入、婚姻状况、和家庭结构。通过聚类分析，Okazaki 发现年轻、富有的单身女性往往对移动电子商务最为积极，在另一方面，受过良好教育的专业人士往往是最积极的。这可能也是移动电子商务发展缓慢的原因之一。因此，研究移动商务用户的认可时，我们不应该忽视用户的人口统计数据。我们建议结合性别和年龄等因素，形成单一的因素用户人口统计，同时在修改后的模型中应当包括职业，收入及婚姻状况。

IT 采纳研究一直被批评太关注感知使用行为，并没有对系统自身的满意度给予足够关注，因此，修改后的 UTAUT 可以延伸至包括用户满意度。为了解 IT 采纳研究，在 IS 领域进行用户满意度研究是非常有用的。根据用户满意度理论，它由许多构建体来确定，它由系统质量，信息质量，以及相关的服务质量所组成。Wixom 和 Todd 将满意度定义为用户对待信息系统自身、系统质量和信息质量的态度，这些都是基于对象的态度和理念。在 IT 采纳方面，基于对象的态度和理念通过使用意图，实用性及易用性来反映，具体来说，对于系统的用户满意度越高，用户对系统易用性的感知就会越高。因此，我们可以通过引入如系统质量和信息质量等系统特点到 IT 采纳框架，从而将用户满意度的研究成果整合到 IT 采纳验证中。如 Wixom 和 Todd 的经验证据表明，这些系统特性的纳入将增强我们在 IT 采纳研究领域预测用户行为的能力。

由于以有线互联网为基础的电子商务和移动电子商务在用户界面，传输速率，输入和输出，信息丰富度和网络兼容性方面具有显著差异，移动电子商务用户往往对系统的功能更为敏感。因此，我们认为有必要在用户接受度调查中包含移动信息系统功能。系统功能包括用户对系统满意度和信息满意度的态度，系统满意度由可靠性，及时性和集成性组成，信息满意度由时效性，准确性，格式和完整性决定。在这一方面，Ives et al, Melone, DeLong, McLean 提供了坚实的理论基础，但是，在应用这些概念时，需要为不同类型的移动电子商务应用作进一步的调整。

信息技术不是中立文化，同样，移动电子商务的发展很可能被一个国家或地区的社会，经济，文化背景所影响。例如，NTT DoCoMo 的成功是唯一且不可复制的，日本独特的文化是 iMode 爆炸式增长的原因之一，没有 IT 采纳理论能解释这种迹象，日本的独特文化是一种可能的解释。对研究者来说，在 IT 采纳理论研究中包含文化因素很不寻常，这些研究的目的是探讨文化对用户的采纳决策的影响，并研究在不同的国家采用 IT 理论的适用性。例如，Straub 等人发现在日本与北美之间，TAM 理论的正确性大打折扣。在类似的研究中，Harris 等人发现虽然英国和香港都具有相同的无线通信基础设施，但用户对移动电子商务的态度和行为大有不同。许多实证基础的研究证据表明，文化因素将影响独立变量用户接受度，因变量及关系缓和。

为了在中国研究移动电子商务用户接受度，文化将被考虑为一个重要的调节因子，应该被添加到修改后的 UTAUT 模型中，使模型更适用于中国国情。为此 Hofstede, Martinsons, Westwood, Srite, Karahanna 为我们提供了确定和衡量中国文化的理论基础，除了这些作为权利差距的传统文化

维度，例如不确定性规避，个人主义和集体主义，男性和女性，我们可能需要考虑为移动商务用户接受研究提供额外的文化结构。

为了在中国研究移动电子商务用户接受度，通过结合移动电子商务的特点，中国文化以及用户满意度等因素，我们已经提出修订 UTAUT 模型。我们建议信任，隐私保护以及成本应包含在修订后的模型中。我们建议引入特定的系统缓和因素，即系统质量和信息质量。

该理论成立后，必须对修改后模型进行测试，报告如下条件：制定正确的研究问题，设计一个适合研究问题性质的研究，开发研究仪器来测量研究模型的构建体，并且收集和分析数据以验证模型。为了验证修改后的 UTAUT 模型，在移动电子商务发展的不同阶段，包括横向，纵向案例研究，关注小组方法等多种方法应被应用于理解用户接受度问题作为移动电子商务的演进。