

硕士研究生学位论文

|  |  |
| --- | --- |
| 题目： | 一个金融网络销售应用 |
|  | 软件的设计与实现 |

|  |  |
| --- | --- |
| 姓 名： | 高强 |
| 学 号： | 1201220692 |
| 院 系： | 软件与微电子学院 |
| 专 业： | 软件工程 |
| 研究方向： | 电子商务与物流 |
| 导师姓名： | 李杰 教授 |

二〇一 六年 一 月

版 权 声 明

任何收存和保管本论文各种版本的单位和个人，未经本论文作者授权，不得对本论文进行复制、抄录、拍照、修改或以任何方式传播等行为。否则，引起有碍作者著作权益之问题，将可能承担法律责任。

摘要

互联网+的时代已经来到，互联网金融已然变成妇孺皆知的热词。2012年前后各种理财平台如雨后春笋般迅速出现，国人的理财选择变得前所未有的丰富。在享受互联网金融发展红利的同时，我们注意到互联网理财的人群呈现出年轻化，互联网理财的交易额呈现出小额高频的特点。大额理财的交易，多数还是通过专业理财顾问或机构来线下交易完成。在此背景下，本文设计并实现了金融网络销售应用软件，将在传统理财方式中扮演重要角色的理财师引入到金融网络销售的应用软件中，给每个注册用户都匹配一位线下理财师，让用户可以在购买过程中得到理财师的专业建议，安心的完成理财。

本文首先分析了互联网金融的产生背景以及国内外类似应用产品的发展现状；接着讲了本项目实现过程中用到的相关技术知识；然后按照软件开发的流程，从市场机会和用户需求角度阐述了金融网络销售应用软件的需求分析，其中用户分析包含了投资用户、门店理财师和平台管理管理员；接着给出了金融网络销售应用软件的总体设计，从系统环境、架构设计到了模块设计；并给出了用户功能的详细设计与实现；最后给出了实现成果和测试结果。

本文中论述的应用软件整体框架是MVC，用到的设计模式是：单例模式，观察者模式。安卓端用到了Intent意图技术、Json数据处理技术、Handler 消息机制、HttpClient 相关技术使用、SharePreference存储技术。Android用到了四大组件：Activity，Service，Content Provider，Broadcast receiver。iOS端用到了objective-c开发语言，最新的sdk。本文的研究对于软件开发公司设计移动端及后台系统具有一定的参考作用。

关键词：互联网金融，应用软件，MVC，移动端开发， MySQL数据库，Client/Server软件架构模式

Design and implementation of a financial net sales

of application software

Qiang Gao ( Software Engineering )

Directed by Professor Jie Li

ABSTRACT

The Internet has numerous traditional industries to the formation of a revolution, the Internet has become a hot word finance including women and children. After 2012, a variety of financial platforms have sprung up rapidly emerging, people's financial choice has never been rich. Online Banking are mostly small urban white-collar financial behavior of large financial transactions, the majority or by a professional financial adviser or institution to the line to complete the transaction. In this context, this paper designs and implements financial network application software sales, financial planner can lead network marketing application software, registered users are matched to each line next financial planner, financial planner so that users can get in the buying process professional advice and secure complete financial management.

This paper analyzes the background of the financial network of sales of application software and similar products at home and abroad, the research objectives of this paper and contributions; followed elaborated key technologies used in this project; then, according to the software development hierarchy of needs, analysis of the financial needs of network application software sales, including investment user, after the financial store and platform management too; then given financial network application software sales overall design, including architecture design, module design, system environment; Finally, the user detailed design and implementation functions.

Application software overall framework discussed herein is MVC, design patterns used are: Singleton pattern, observer pattern. Andrews end technology used in the Intent intent, Json data processing technology, Handler message mechanism, HttpClient related technology use, SharePreference storage technology. iOS end uses objective-c programming language, the latest sdk. Android uses four components: Activity, Service, Content Provider, Broadcast receiver. This study design for software development companies moving end and back-office system has a certain reference.

KEY WORDS: Internet banking,MVC，Mobile-side development, MySQL DB, Client / Server

目 录

[第一章 绪论 1](#_Toc432758729)

[1.1 研究背景与意义 1](#_Toc432758730)

[1.2 国内外现状 2](#_Toc432758731)

[1.3 主要研究目标、内容和贡献 4](#_Toc432758732)

[1.4 论文框架结构 4](#_Toc432758733)

[第二章 相关技术研究 6](#_Toc432758734)

[2.1 MVC模式开发 6](#_Toc432758735)

[2.2 Spring框架 7](#_Toc432758736)

[2.3 MySQL数据库 9](#_Toc432758737)

[2.4 Json数据格式 11](#_Toc432758738)

[2.5 本章小结 12](#_Toc432758739)

[第三章 金融网络销售系统需求分析 13](#_Toc432758740)

[3.1 市场分析 13](#_Toc432758741)

[3.1.1 现有市场分析 13](#_Toc432758742)

[3.1.2 存在的机会 15](#_Toc432758743)

[3.2 功能性需求 16](#_Toc432758744)

[3.2.1 总体功能需求 16](#_Toc432758745)

[3.2.2 功能需求 17](#_Toc432758746)

[3.3 非功能性需求 18](#_Toc432758747)

[3.3.1 应用软件界面与场景需求 18](#_Toc432758748)

[3.3.2 系统应用环境 18](#_Toc432758749)

[3.4 本章小结 19](#_Toc432758750)

[第四章 金融网络销售系统的概要设计 20](#_Toc432758751)

[4.1 系统设计原则 20](#_Toc432758752)

[4.2 系统总体功能结构 21](#_Toc432758753)

[4.2.1 功能框架 21](#_Toc432758754)

[4.2.2 网络拓扑结构 23](#_Toc432758755)

[4.3 系统技术实现架构 23](#_Toc432758756)

[4.4 数据库设计 25](#_Toc432758757)

[4.5 本章小结 39](#_Toc432758758)

[第五章 金融网络销售系统的详细设计与实现 40](#_Toc432758759)

[5.1 金融网络销售用户端应用软件 40](#_Toc432758760)

[5.1.1 注册模块 40](#_Toc432758761)

[5.1.2 登录模块 43](#_Toc432758762)

[5.1.3 项目展示模块 45](#_Toc432758763)

[5.1.4 购买与订单管理流程 46](#_Toc432758764)

[5.1.5 收益资产查看流程 50](#_Toc432758765)

[5.1.6 应用软件设置流程 53](#_Toc432758766)

[5.1.7 交易密码 56](#_Toc432758767)

[5.2 门店管理功能 58](#_Toc432758768)

[5.2.1 我的工作台 59](#_Toc432758769)

[5.2.2 客户管理 61](#_Toc432758770)

[5.2.3 销售管理 70](#_Toc432758771)

[5.2.4 结算管理 73](#_Toc432758772)

[5.2.5 理财师课堂 73](#_Toc432758773)

[5.3 平台管理功能 75](#_Toc432758774)

[5.4 本章小结 76](#_Toc432758775)

[第六章 系统实现展示与测试 77](#_Toc432758776)

[6.1 系统实现 77](#_Toc432758777)

[6.1.1 平台管理实现 77](#_Toc432758778)

[6.1.2 用户端实现 79](#_Toc432758779)

[6.1.3 门店管理的实现 85](#_Toc432758780)

[6.2 系统测试 87](#_Toc432758781)

[6.2.1 应用软件测试 87](#_Toc432758782)

[6.2.2 后台功能测试 88](#_Toc432758783)

[6.3 本章小结 90](#_Toc432758784)

[第七章 总结与展望 91](#_Toc432758785)

[7.1 课题总结 91](#_Toc432758786)

[7.2 不足和展望 91](#_Toc432758787)

[参考文献 93](#_Toc432758788)

[致谢 95](#_Toc432758789)

# 第一章 绪论

## 1.1 研究背景与意义

2013年余额宝的爆发向我们展示了互联网金融的威力：面向普罗大众、通过互联网销售的单只货币市场基金轻松超过了很多基金公司管理资产的总和[1]，这在以前被认为是毫无可能的事,但在这个互联网+的时代的确发生了。作为应对，传统金融机构加速互联网化，互联网创业公司也表现可圈可点，融资金额达到了创纪录的新高。

13年也顺理成章地被视为中国互联网金融的具有里程碑意义的一年，各类媒体大篇幅的报道了互联网金融。但是媒体报道的内容中主要提到的是p2p[2]，其实这只是其中一种互联网金融形式。从广义上讲，任何涉及到了金融服务的相关互联网服务,都可称作互联网金融，包括但不限于众筹、金融中介、P2P（个人对个人）、第三方支付以及互联网理财等模式[3]。

互联网金融模糊了传统金融服务VIP的概念，使得投资1元钱的用户可以享受到1亿元用户所享受到的一些服务，因而使得旧的金融规则被动开始变革。服务于“anytime, anywhere, anyone”的移动支付打破了金融的时间、地域和信息相对封闭的限制，创造长尾客户的金融需求，最有机会实现快速突破，从而抢占传统金融的份额。由国家统计局公布的年度数据显示，居民人均可支配收入一直维持增长，但增长率下滑，间接提升了老百姓对手头现金资产增值的投资需求。

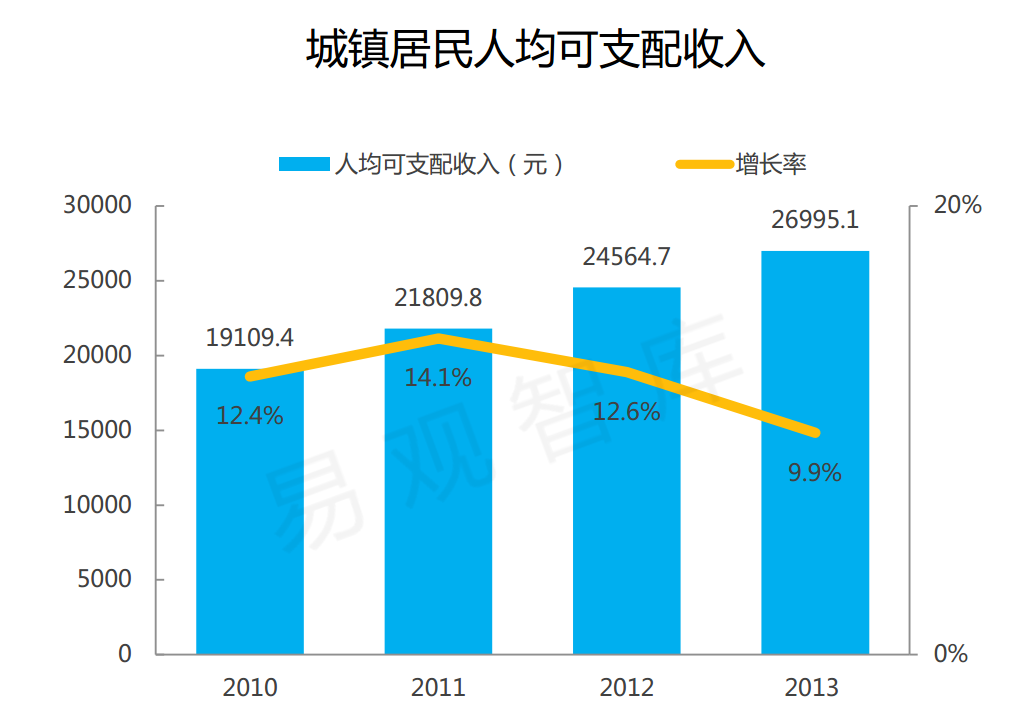


图1. 1 镇居民人均收入

到2014年12月，有7849万网民购买过网络理财产品，较两个季度前增长1465万人。在网民中使用率为12.1%，较2014年6月使用率增长2个百分点[4]。网络理财产品在其发展之初的一年里风光无限，但转入2014年下半年，已基本结束了其用户规模爆发式增长的态势，增速开始放缓，同时新产品扩容速度也有所放慢，主要归结于以下原因：第一，由于网络理财产品的选购方便、高流动性和其发展之初的高收益等特点使其潜在用户已很大程度得到转化；第二，现在市场资金逐渐宽裕、货币基金的收益率持续下降[5]；第三，2014年下半年中国股市的回暖，分流了部分原本流向互联网理财产品的资金，这也对互联网理，产品用户规模的持续快速增长造成一定影响。

综上，互联网金融目前到了一个新的阶段，摆脱了最初毫无限制的增长阶段，现在只有引领创新，才能获得进一步的发展空间。分析现在的互联网理财竞争格局，我们可以发现互联网金融发展到现在虽然已经较为充分，但主要针对的是40岁以下年轻的用户，且用户主体的收入属于中等。对于40岁以上的高收入人群，目前所存在的金融网络销售平台并没有覆盖到他们。这部分人群依然在采用传统的线下理财方式来管理自己的资产。在此背景下，本文设计并实现了金融网络销售应用软件，并将在传统理财方式中扮演重要角色的理财师引入到网络销售的应用软件中，给每个注册用户都匹配一位线下理财师，让用户可以在购买过程中得到理财师的专业建议，安心的完成理财。

## 1.2 国内外现状

互联网金融群雄逐鹿的局面,主要由传统金融机构和互联网公司等其他非金融机构构成。传统金融机构主要是传统金融服务网络的互联网化及电商化,非金融机构主要是指利用互联网技术进行金融运作的电商平台、P2P模型的网络借贷平台、众筹网络投资平台以及第三方支付平台。

其实互联网金融的发展由来已久，贸易加支付做大金融的例子，第三方支付（支付宝们）并不是第一例。17 世纪的荷兰，中国山西票号的兴起，皆从此规律出。2008年初，当美国知名个人金融服务公司银率网悄悄进入中国时，并没有引起大家的特别关注。那时大洋彼岸的P2P网贷企业Prosper和LendingClub还在为是否向美国证交会注册而纠结，它们在中国的模仿者拍拍贷也还默默无闻。支付宝尽管已为大家所熟知，但是在大多数人眼里它只是方便大家网购的支付工具，媒体上“互联网金融”这个词还不常见。5年过后，余额宝首次进入公众视野，首次爆发就给我们展示了互联网金融的威力：面向草根的、通过互联网销售的单只货币市场基金轻松超过了很多基金公司管理资产的总和，现在使用余额宝的用户已接近亿级规模[6]。余额宝的发售成为了互联网金融行业发展历程上的一个重要节点，随后各种互联网理财产品如雨后春笋版的涌现出来。如票据理财产品“京东小银票”，团购金融产品的“百度百发”，保险理财产品“网易收益保”，跨界消费金融产品“百发有戏”等。由此带动了互联网企业纷纷进入金融行业,直接推动了金融脱媒的进程。

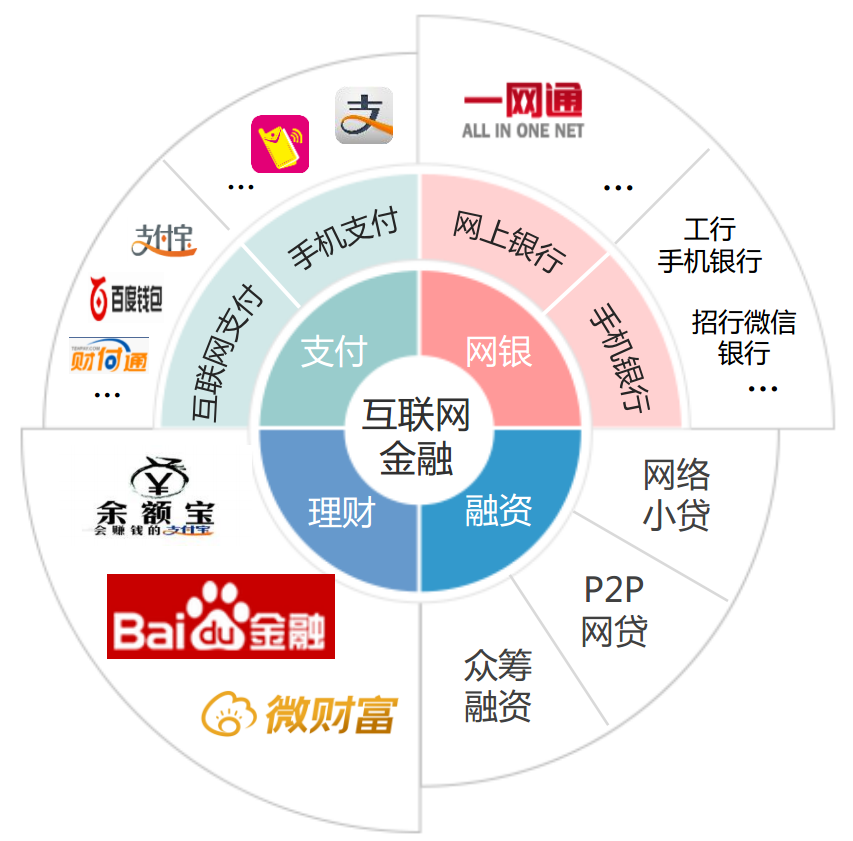


图1.2 热门互联网金融产品

互联网金融的井喷式发展与互联网信用业务的发展息息相关。互联网信用业务使得资金供需的双方基于网络信任机制在互联网上就完成资金融通[7]，是金融互联网化的重要因素[8]。外国经营理念，坚持以信息中介和网上商务平台作为信息中介，以提供完成的服务，收取服务费的原则，不干预其自有资金的借贷关系，同时，所有的信息披露，销售，交易和风险控制等所有在线实施，从而实现了脱媒[9]。以P2P为例，最初是诺贝尔和平奖得主尤努斯教授最先提出了P2P模式的贷款，英国人在05年第一个推出网贷平台Zopa，经过10年的发展,目前已经成为交易量第一平台。在国外做的比较好的还有巴西的Fairplace、韩国popfunding、德国的Auxmoney、日本的Aqush、西班牙的Comunitae、冰岛的Uppspretta等等[10]。从上面的论述中，我们可以预见，互联网金融将带来巨大的改变,将彻底转换现存银行的运作方式[11]。

## 1.3 主要研究目标、内容和贡献

现在互联网金融发展的如火如荼，很多人深切的感受到了这次浪潮的红利，但也有更多的二三线城市的人群，由于种种原因并没有参与进来[12]。在此背景下，本文设计并实现了金融网络销售应用软件，可以将理财师引入网销应用软件，给每个注册用户都匹配一位线下理财师，用户在购买时产生的疑惑可以和理财师沟通，从而得到专业的建议。并且在抢购一些优质的资产时，理财师可以帮助用户提前锁定资产，方便用户抢购。

整个金融网络销售系统产品功能设计的都是围绕 “帮助用户实现投资行为”这一产品目标展开，并以此目标作为项目边界与项目效果的评估标准。手机是主要的设计场景。初期不特别明晰平台间的交互区别，统一设计主体功能、业务逻辑与视觉界面，追求多平台的体验统一。在这一目标达到后，再针对各个平台的实际情况做具体的体验细节修改，以形成各平台的产品方案。理财师是通过PC的CRM来协助用户进行购买操作。

整个金融网络销售系统指某互联网金融企业的投资客户和理财师使用的产品集合体，用户端包括注册与登录流程、项目展示流程、购买与订单管理流程、收益查看流程、支付与提现流程、应用软件设置流程等。理财师管理后台包含我的工作台、客户管理、销售管理、员工管理、权限管理、理财师学院以及消息模块。

## 1.4 论文框架结构

本文介绍了某互联网金融企业作为一个平台，包含了项目方、理财门店和用户，为使三方的共同协作，建立了应用软件、理财门店管理后台以及平台管理后台。他们的需求各不相同，但却都有联系。文章结构如下：

第一章、绪论：介绍了课题背景、研究目的与意义、国内外互联网金融的发展现状，分析了金融网络销售的发展现状，以及能够去创新的地方。

第二章、相关知识介绍：本章介绍了设计与实现金融系统会用到的技术，在后面的系统架构及实现中会提到应用，主要包含了MVC模式开发、Spring框架、MySQL数据库、JSON数据格式。

第三章、金融网络销售系统的需求分析：本章主要分析了金融网络销售系统的需求来源，并把需求拆解成了用例，用面向对象的方法进行了详细分析。本章同时提到了系统所需要的环境，包含了服务器和移动终端的设计被要求。

第四章、金融网络销售系统的概要设计：主要讲了系统设计时参照的原则，以及系统的架构设计方案，然后从完整的业务组成上分析了金融网络销售的用户功能框架和门店管理用户框架，最后讲了数据库的设计以及产出的一些主要的表。

第五章、金融网络销售系统的系统详细设计及实现: 本章分流程讲了金融网络销售系统的设计方案以及实现手段。分成三个 大的部分来讲，用户应用软件端、理财门店端以及平台管理后台。每一部分都按业务流程分开来称述。

第六章、系统成果展示与测试：本章对最终实现的成果进行展示，其中包括了注册登录、项目展示、下单支付、生成预售码、客户管理等主要流程，最后对系统进行了测试。

第七章、总结与展望：总结贡献、创新点与不足之处并展望了未来的研究

# 第二章 相关技术研究

## 2.1 MVC模式开发

MVC是众所周知的设计模式，其中包含了23 GOF设计模式及其优化版本[13]。MVC本质是三部分构成的，即模型视图控制器组成的，其中，Model是程序的核心，表示软件的数据库相关信息等，View显示数据库记录，Controller处理写入到数据库中的记录 [14]，MVC将业务逻辑，数据组织，界面显示的代码点分离，聚集在一个业务逻辑组件内，在改善和个性化的同时自定义界面和用户交互，不需要编写业务逻辑[15]。 MVC开发独特地用在图形用户界面的逻辑结构映射传统输入，处理和输出功能。MVC开始在桌面应用程序存在，M代表的业务模式，V是指用户接口，C是控制器，使用MVC将模型和视图的属于不同的模块[16]，使相同程序拥有多重表现方式。如一些统计数据可以通过柱状图，饼图分别表示。 C的目的是确保M和V同步的，一旦M变化，V应该有一个更新。MVC模式结构图如下图2.1所示：[19][20]

Control

用户输入

改变显示

改变状态

获取数据

Model

View

通知视图

图2. 1 MVC模式结构图

MVC模式的工作流程可以概括为以下几点：

1. 客户端浏览器向服务器提出请求。
2. 应用程序向Server端发出请求，接受到的请求被提交给控制器处理[16]。
3. 控制器处理请求。控制器处理请求的过程一般包括执行情况的检查及（创建）模式。执行条件的检查装置，一些输入项，确定该请求是有效的，例如身份验证。控制器的设计要求复用性要高[17]。应对与每个应用程序的具体细节相关联的请求，可以通过模型来完成，建立一个模型，并提供服务调用模型。数据通常存储在持久层。
4. 视图接受控制器转发的请求。
5. 模型提供相关服务，视图进行调用调用[18]，这其中主要处理的是数据相关的问题。
6. 客户端一直在等待相应，视图整理好数据后发回到客户端[19]。

MVC模式将M和V分离,从而达到了便捷和提高复用率的效果，如果没有采用分离的处理,用户界面设计往往要耦合这些对象[20-22]。把模型和视图分离开，具有以下优点：

1. MVC模式使得程序的结构更加清晰，编写代码的程序人员也可以分工去处理不同的业务，之间不受其他的影响。
2. 代码按一定的规律模块化后，会变得更加健壮，稳定性更强[23]。
3. 用户界面可以改变很快。模型与视图一对多，当模型内容变化，视图将得到迅速更新[24]。
4. 模块化之后，模型可以单独的移动到其他地方复用，这直接提高了程序猿的工作效率。
5. 提高开发效率。MVC将代码分成不同的模块，这样可以使得更加专业的人员去处理擅长的部分，比如设计UI、业务组织 、数据准备都会有专业人员去处理[25] [26]。

## 2.2 Spring框架

Spring是一个开源框架，最初由哈罗德·约翰逊提出。他的提出给开发企业级应用带来了福音[27]。使得EJB来执行的事情，现在可以由Spring使用基本的JavaBean来完成[28]。然而，Spring不限于服务器侧的开发。从可测性，简单性和松耦合，所有的Java应用程序可以受益于Spring。因而可以说，通过控制（IoC）和面向（AOP），Spring很好的扮演了一个轻量级框架的角色。

（1）量级：无论从大小和成本来衡量，Spring都可以被定义为一个轻量级的框架，核心完整的Spring框架可以在一个JAR文件的大小超过1 MB的文件里

（2）Spring通过一个被称为控制反转的技术，促进了松散耦合。控制反转，与此相反的JNDI中，不从容器对象找到关系，但容器初始化时，从对象请求将依靠活性传递给它。

（3）面向方面：Spring提供了广泛的支持面向方面的编程。而应用对象只实现了他们应该做的，就是完整的业务逻辑。如日志或事务支持等系统级问题，不是它们来操作的。

（4）容器：应用对象的增删改查相关配置都是有Spring来掌控，包含并管理的配置和生命周期可以把它理解为一个小型的仓库，但是，应该记住spring和重量级的EJB是显然不同的，后者经常是大而笨重的，晦涩难用。

（5）框架：Spring通过搭积木一样，把组件配置组合成为新的应用程序。Spring支持很多常用的事件（Task管理，Persistence framework integration等等），你只需要注意的应用程序逻辑[29-32]。

所有这些功能使代码更美观、更健壮、测试也会更加方便。

Spring框架由7个模块，各个模块定义清晰，各司其职，具体如图2.2：

Spring AOP

资源层次

AOP下层构造

Spring ORM

Hibernate 支持

JDO支持

Spring WEB

WEB应用内容

多点解决方案

Spring WEB

MVC

WEB MVC 架构

WEB查看器

JSP/Velocity

PDF/EXCEL

Spring DAO

JDBC 支持

DDO支持

Transtraction 下层构造

Spring Context

应用内容

UI支持

JNDI，EJB&

Remoting Mail

Spring Core

支持效果

Bean容器

图2.2spring框架

Spring框架包含的7个模块，可以独立运行，也可以好几个模块实现更复杂的功能。各模块功能描述见下文：

1. 核心容器：BeanFactory做为最要的部分，可以提Spring的常规功能，通过BeanFactory适用的应用程序配置控制（IOC）模式反转和依赖性规范与实际的代码中分离出来。
2. Spring 上下文：这个配置文件包含了上下文信息，同时还内含了企业服务。
3. Spring AOP：运用Management and Configuration，面向方面的编程借助这个模块被嵌入到了Spring框架中。AOP的Spring框架可以兼容任何对象，事务管理服务可以放在Spring AOP模块可以进行。
4. Spring DAO： 异常层次结构通过抽象层而存在，这个结构来可以用来管理异常处理，也可以用来处理相应数据库提供方的。该异常非正常消息。Error的处理由于异常层次结构的存在变得容易，并且使得异常代码的工作量少了很多，即你需要编写（如打开和关闭连接）的数量。
5. Spring ORM： ORM框架嵌入在在Spring框架，它提供对象的关系， Spring常规事务和DAO异常层次结构包含了所有这些关系工具。
6. Spring Web ：它的作用是提供web的背景信息,它本身是依赖于应用程序的背景关系。所以，Spring框架和Jakarta Struts能够组合在一起使用。待处理部分及参数捆绑到域对象的工作也通过这部分模块得到了简易化处理。
7. Spring MVC ：这个框架可以用来实现完成的web软件。MVC 框架运用策略API后，配置的灵活度非常高，众多视图技术也被包含在了MVC中[33-35]。

Spring框架的应用的场景很广泛, Java 2 Platform Enterprise Edition及其变种服务器都可以使用。Spring的要点是：可以不捆绑到指定J2EE服务的可再用业务以及数据访问对象。显而易见的是在不同的J2EE环境，测试环境之间以及独立应用程序中可以多次调用此对象[36]。应用Spring框架会带来如下益处：

1. 可以有效的将中间件组装起来
2. 将能够消除Singleton不合时宜的使用
3. 在Spring建立一个统一的配置信息，消除各种自定义格式属性
4. 实现真正意义上对接口编程。
5. 在大多数Spring应用程序的业务对象，没有依赖于Spring。
6. 单位测试容易展开
7. 支持JDBC和O / R映射产品
8. MVC Web框架，清晰的MVC实现，它将提供一种非侵入性的方法。
9. 很容易改变的实施细则，JNDI抽象层，它可以让您轻松切换远程和本地服务。
10. 简化访问数据库的异常处理。
11. Spring直接就可以管理事务，不是通过够操纵GTA
12. 可以支持JavaMail以及其他一些主流的邮件系统。

## 2.3 MySQL数据库

MySQL是一种基于关系来生成的数据库，MySQL AB是这个伟大产品的缔造者。MySQL的Web方面的应用是最好的RDBMS解决方案之一。MySQL根据存储在不同的表，而不是把所有的数据堆砌在大型仓库[37]，从而增加了查询，添加，删除，修改，速度和表和表之间的关系，以提高数据库的灵活性。SQL作为操作数据最流行的语言，开源的特性让其越来越完善，MySQL用的就是这种语言，很多规模不大的公司在设计数据库时通常选择MySQL作为数据库。相比那些大型数据库，如Oracle 、DB2、SQL Server， MySQL有其不足之处，但是这并不能减少它的受欢迎程度。很多初创的公司以及规模不大的公司用MySQL提供的功能是绰绰有基本可以满足需要，并作为MySQL的L是开源软件，因此可以大大降低总体使用费用。

Linux是常用的操作系统，Apache和Nginx共同组成了Web Server，MySQL充当Datebase,PHP/ Perl/ Python充当server端的Script Interpreter。这四个软件都是开源不收费的（FLOSS），因而用这个方法可以不用花钱去买专利（除人工成本），就可以了搭建一整套web系统，因为IT从业者把这几个软件合称为“LNMP”组合。

MySQL 有很多优良特性

1．支持常用的OS，主流的都包含在了其中。

2．支持主流语言调用API。你能想到的基本都包含其中[39]。

3．多线程运行，使得CPU的资源不被浪费。

4．查询算法不断的被优化，查询效率比较能接受。

5．使用比较灵活，可以单独使用，也可以嵌入来使用，如作为一个库。

6．可选择的语言种类很多，主流的语言都得到了支持。

7．接入到数据库方式选择很丰富。

8．提供用于相应的数据库操作的工具。

9．支持海量数据的数据库。

除了以上Mysql的特性以为 它也为开发者提供了许多便捷的功能。

1.数据库的备份与还原，这样可以轻松的完成数据库的移植。

2.文本转到数据库，按着一定的规则将文本数据组织好，这样就可以轻松的将一个文本格式的数据存放在数据库表中。

3.数据视图，可以轻松完成复杂的数据查询，在应用调用的时候将会是很方便的。

4.存储结构，支持复杂的SQL语句，支持多表联动[40]。

5.索引，可以更快捷的查询数据。

MYSQL用于互联网的优势

1. 借鉴互联网产品，例如淘宝，底层的关系数据库都是使用mysql，特别是数据魔方，底层就是MYSQ。

2.开源数据库支持多钟语言调用API。

3. mysql支持不同的存储引擎，可以根据业务的需要选择不同的引擎。例如：支持事务处理的InnoDB存储引擎提供了具有提交、回滚和崩溃恢复能力的事务安全。 所以对于互联网产品，响应速度将决定着一个产品的成败，据数据显示，网站每延迟一秒将会造成用户满意度下降16%，甚至做成每年250万的损失。 而mysql的myISAM引擎不支持事务处理，但是响应速度是非常快的，特别是5.5版本之后，响应速度有了巨大的提升，所以，可以根据业务的不同需要进行选择不同的引擎。

4. 讲到互联网产品，大多是读写差距比较大，Mysql提供了简单的管理操作并且易于维护，使用数据复制功能就可以非常容易的实现主从读写分离，实现负载均衡与mysql的集群，同时，mysql是表级锁，当写入一张表操作时，其他的线程将无法在操作这张表，所以主表只负责写操作，多个从表实现读操作，提高系统的读写速度。

## 2.4 Json数据格式

JSON (JavaScript Object Natation)是一种基于纯文本的数据交换格式，通过对象和数组嵌套组合以表示各式各样的数据，和XML比较，其可以借助javascript语言存储复合对象，并且有效信息交互存储效率更高、读写更快、且没有保留字等限制。其特点是：轻量、灵活、使用方便。JSON标准(RFC4627)于2006年7月发布，核心结构有：

对象：

{"city":"Shanghai","street":" Guanglan Road ","postcode":200120}

对象嵌套：

{"name":"Michael","address":{"city":"Shanghai","street":"GuanglanRoad","postcode":200120}}

数组：

[{"firstName":"Pluto","lastName":"Smith"},{"firstName":"Bush","lastName":"Leo" },{ "firstName":"Tom" , "lastName": "Carter" }]

JSON的应用非常广泛，Javascript语言中使用JSON非常方便，其内建eval()函数，可以直接将JSON数据源生成JS对象。例如：

var data={"city":"Shanghai","street":" Guanglan Road ","postcode":200120}

var dataObj=eval(“(“+data+”)”)

前端开发人员可以进入Json2.js库，该库对浏览器的兼容性非常好，用于不在支持JSON对象的浏览器使用（IE7及以下）。使用方法：

序列化,即Json对象转String：

var jsonObj = { id: '01', name: 'Tom' };  
JSON.stringify(jsonObj);

反序列化,即String转Json对象:

var jsonString = "{ id: '01', name: 'Tom' }";  
JSON.parse(jsonString);

在B/S应用中，通过Ajax进行客户端和服务端交互（JSON-RPC over http所用到的就是JSON，使得应用适应性更好，能够轻松的对接来自WEB，IOS和Android的请求，并且解析效率更高。当然，JSON的应用不仅体现在交互中，也在存储上发挥了一定的影响力。在WEB应用，开发者在关系数据库中把JSON对象作为一个字符串的形式来存储，以此更好地和更方便地保持对象的全部信息。在Java应用程序中，用户也可以运用JSON解析序列化和反序列化Java对象。目前，java语言json处理主要解决方案有：json-lib，gson，jackson，fastjson等。在使用Ajax技术，经常会序列化处理Java对象， GSON在应用程序使用来处理这类情况。 Gson对象是解析的最基础工具类，主要有两个方法toJson()序列化，fromJson()反序列化，例如：

String[] days = {"Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"};

Gson gson = new Gson();

String json = gson.toJson(days);

获取该对象的方式主要有两种：

1、new Gson()

2、new GsonBuilder().create()

Gson也提供了对复杂对象（或泛型）的处理能力，其中JsonParser类负责解析析JSON到JSON元素的解析树，TypeToken类用于创建type，或者带复杂泛型的List和Map。

JSON应用中，为了方便开发人员，恰当使用某些工具非常有意义，例如Java字符串转义工具、JSON字符串生成工具、JSON格式化校验工、JSON数据视图工具以及模拟REST请求工具等。

## 2.5 本章小结

相关知识介绍：本章介绍了设计与实现金融系统会用到的技术，在后面的系统架构及实现中会提到应用，主要包含了MVC模式开发、Spring框架、MySQL数据库、JSON数据格式。

# 第三章 金融网络销售系统需求分析

## 3.1 市场分析

### 3.1.1 现有市场分析

互联网金融的主要类型包含众筹、第三方支付、p2p贷款以及互联网理财等类型。其中较为大众所熟知的是p2p贷款和互联网理财，截止到本文撰写，据网贷之家统计的数据来看，全国共有网贷网络借贷平台2283家[41]。

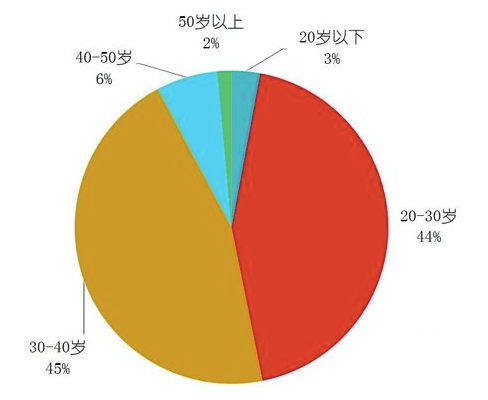


图3. 1 互联网理财用户年龄分布

由上图可以看出，互联网理财用户中，青年人居多，20-40岁的投资用户占比高达89%。从下图我们可以看出85%的互联网理财用户年龄在40岁以下，中国额互联网理财用户，年龄偏年轻化。不难理解年龄更大一点的用户对互联网技术的掌握度很低，因而对互联网金融的红利得到较少。

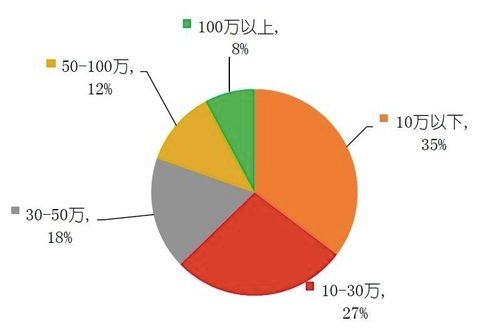


图3. 2 理财用户的收入分布

如上图，互联网理财的用户中，年收入在30万以下的占比62%。50万以上的大额投资只占了总投资人数的20%。更多的中产阶级其实还是在用传统的理财方式，这些高净值用户是任何网贷平台的种子用户。

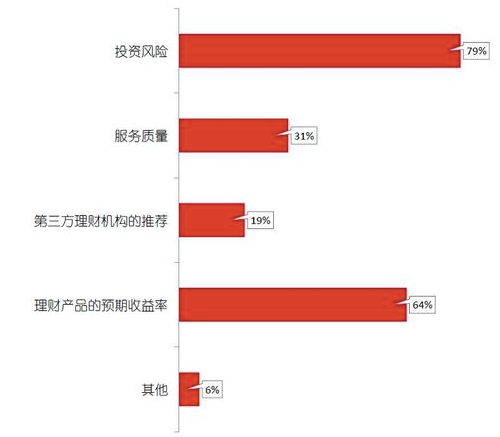


图3. 3 影响决策的因素分布

从上图可以看出，影响用户是否进行互联网理财师的最主要的因素是收益率和投资风险。服务质量也是一个相对重要的因素。互联网金融的核心竞争力就在与超高的利率、快速且高效的服务，最重要的一点是做好风控，良好的项目质量是网贷平台能否发展起来的关键。



图3. 4 中国手机网名规模及占网名比例

到2014年12月，我国上网的人数达6.49亿，14年新增网民3117万人。我国通过手机上网的人多达5.57亿，相比13年增加了5672万人。所有接入互联网的人群中,通过手机来联网的人群占比从2013年的81.0%提升至85.8%。[42]通过进一步调研发现71%的用户更倾向于用移动客户端来完成操作，只有29%的用户任务PC的稳定性会高于移动端。不可质疑的市移动端的便捷和易操纵性，已经不断解决的安全保护问题，让pc的优势变得不那么明显。因此在移动端的优先级高于PC。

### 3.1.2 存在的机会

现在互联网理财虽然已经发展的较为充分，但主要针对的是40岁以下年轻的用户，且用户主体的收入属于中等。对于40岁以上的高收入人群，目前所存在的金融网络销售平台并没有覆盖到他们。这部分人群依然在采用传统的理财方式。

在此背景下，本文设计并实现了金融网络销售应用软件，主要在以下三点有所不同：

第一，引入理财师。通过LBS为给每个注册用户都匹配一位线下理财师，让用户可以在购买过程中得到理财师的专业建议，安心的完成理财。平台签订的理财师是通过渠道的额同学在线下拓展而来。理财师从事外表光鲜的职位，但依然用着最原始的方法记录和管理着自己的工作，他们亟需一款销售工具，在这个工具中，可以协助他们完成资料的记录、工作方法的累积，让理财师可以快速的进阶。从而达成技术提高效率的目的。

第二，一键购买的流程。针对用户比较忙碌和对互联网的接受度有限的用户特点，为这类用户设计的软件一定是简单可上手，因而在设计软件时应在设计和交互上更注重中老年用户群体的用户习惯。

第三，更优质的资产。从上小结的分析我们可以看到，在购买时，资产的安全考虑是决定用户是否购买，购买多少的重要因素。因而在筛选资产时，风险控制显得尤为重要。业务部门设计了四道审核：项目方自审、法务审核、金融侧风控审核以及最后的财务审核。

总结：虽然现在网贷平台很多，但是缺少一个精准满足高端理财需求的平台，这些用户尚未被发掘和满足，这是一个新的机会。

## 3.2 功能性需求

### 3.2.1 总体功能需求

某互联网金融企业作为一个平台，包含了项目方、理财门店和用户。三方的共同合作使得平台顺利运转。用户先进行注册并浏览理财产品，然后通过银联充值，可以选购理财产品，用户也可以发起提现等行为，用户在浏览理财产品的过程中可以咨询理财师相关问题。理财师自己的CRM后台可以看到新注册用户的相关信息，如果用户需要购买预售的理财产品还需要对理财师索取预售码，因此理财师也可以主动把握销售机会去联系用户。在他们的需求各不相同，但却都紧密联系，接下来我们分别分析他们的需求出发点。

### 3.2.2 功能需求

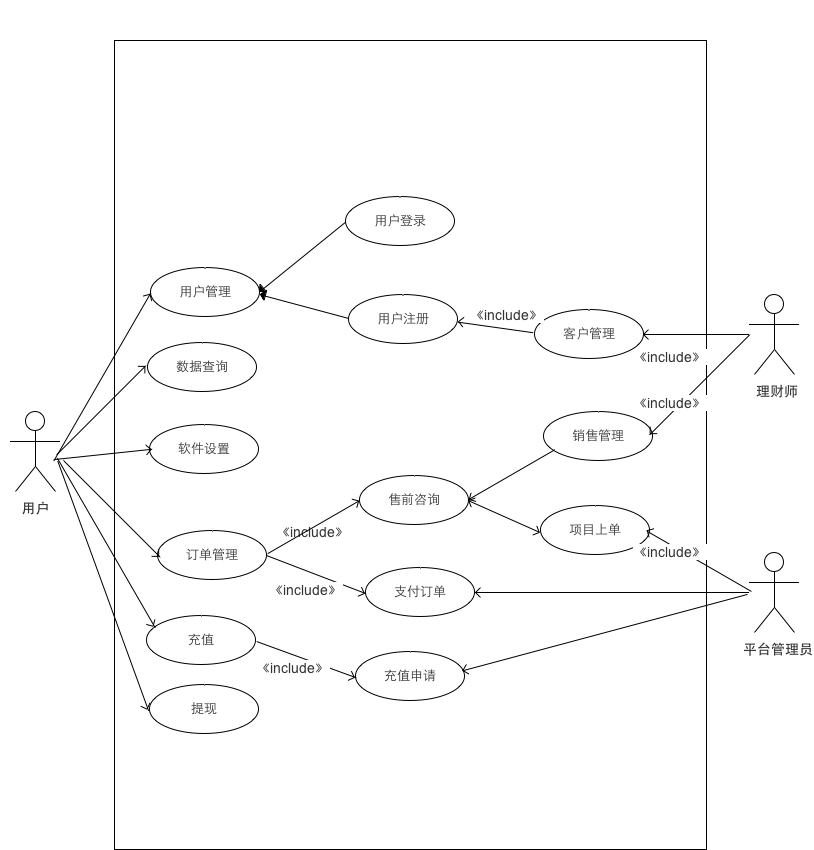


图3.5 金融网络销售系统用例图

用户端包括注册与登录流程、项目展示流程、购买与订单管理流程、收益查看流程、支付与提现流程、应用软件设置流程等。C端的移动端产品包括android 应用软件、IOS 应用软件和mobile-web 3个板块。

理财师理财师是平台和用户接触最多的一个角色，理财师管理后台包含我的工作台、客户管理、销售管理、员工管理、权限管理、理财师学院以及消息模块。通过客户管理和销售管理，理财师可以把握用户的信息，从而更好的抓住销售机会。

平台管理员通过audit自用系统进行系统维护，同时用来实现对交易进行中间操作。例如用户进行线下充值需要平台进行审核，审核通过后才可以进行账户余额的变更；提现操作，用户在客户端发起提现操作，由后台管理人员进行审在管理后台可以看到用户发起的申请，财务人员进行核查，对于核实无误的申请进行第三方打款操作；平台需要上新的项目时，需要在后台进行上单操作，填写项目的信息，并进行财务、风控和法务三道审核。因而audit中包含了提现审核、门店管理、项目管理、充值审核等功能。

## 3.3 非功能性需求

### 3.3.1 应用软件界面与场景需求

解决用户亟需：解决用户亟需是实用性的提现。

规范化平台:一个能够广大发展的平台一定是一个规范的平台，各种服务和第三方接口的接入，会给平台带来的源源不断的生命力。

完成的数据：作为一个金融网络销售的平台，直接发生现金流，用户的利益息息相关，因而完整的数据，不可随意修改的数据保存显得尤为重要。

### 3.3.2 系统应用环境

平台采用的是阿里云的服务器,阿里云是[阿里巴巴集团](http://baike.baidu.com/view/1247049.htm)旗下云计算品牌。用户通过阿里云，用[互联网](http://baike.baidu.com/view/6825.htm)的方式即可远程获取海量计算、存储资源和大数据处理能力。截至2014年6月，在阿里云购买云服务的用户已过百万，遍布互联网、移动应用软件、音视频、游戏、电商等各个领域。根据IDC调研报告，阿里云是国内最大的公共云计算服务提供商。某金融企业共购买5台服务器，参数及用途如下所列。

**服务器配置**

表3. 1 服务器配置

|  |  |  |  |
| --- | --- | --- | --- |
| 实例名称 | 公网IP | 内网IP | 配置 |
| frontend\_srv1 | 101.200.201.93 | CPU:2核  内存:2048 MB  带宽:4Mbps | 前端网站部署， 包括pc/crm\_pc/touch服务 |
| backend\_srv1 | 123.57.241.242 | CPU:2核  内存:4096 MB  带宽:2Mbps | 后端服务器，部署后台服务模块， 此机器上部署deal,  passport, order，  demo\_crm\_pc服务和redis缓存 |
| backend\_srv2 | 123.57.242.136 | CPU:2核  内存:4096 MB  带宽:2Mbps | 后端服务器，部署后台服务模块， 此机器上部署 pay, crm |
| NFS | 123.57.241.242 | CPU:2核  内存:4096 MB  带宽:2Mbps | 此服务器为图片分享服务器， 也可放一些共享的静态文件， 同时还部署jenkins和maven  的nexus repo 署后台服务模块， |
| Develop | 123.57.206.225 | CPU:2核  内存:4096 MB  带宽:2Mbps | 此机器为测试机，所有服务 模块均在此机器上部署一套， 以供正式发布前的开发测试用 |

**移动终端要求**

本文中的应用软件有安卓和iOS版本。安卓系统千差万别，安卓的生产厂稂莠不齐，想要做到所有的机型面面俱到是不可能的。因而我们选取了安卓主流的尺寸1920\*1080.这样在大部分主流机型上都可以正常显示。某互联网金融企业应用软件支持安卓2.0及以上系统。

IOS客户端支持ios6.0及以上系统截止到本文撰写时iOS系统已经更新到了8.4.1。某互联网金融企业的所有涉及搞对尺寸不一的机型做了专门的适配，对市场上存量极大的iPhone4s及iPhone5、iPhone5c、iPhone5s、iPhone6、iPhone6s都具有良好的兼容性。

## 3.4 本章小结

本章主要分析了金融网络销售系统的需求来源，市场现状，并把需求拆解成了用例，用面向对象的方法进行了详细分析。本章同时提到了系统所需要的环境，包含了服务器和移动终端的设计被要求。

# 第四章 金融网络销售系统的概要设计

上一章主要阐述了系统的设计原则及设计思路，并且分析了系统中主要角色的相互关系及现实需求。这章将围绕系统的功能框架、数据库设计以及系统安全的考虑。

## 4.1 系统设计原则

**可靠性**

当系统复杂到一定程度,要保证良好的可靠性就需要付出更多的成本。在该系统的可靠性的应用本身运行越来越高，该软件系统的可靠性，直接关系到设计自己的信誉和竞争[43]。软件可靠性意味着软件测试中表现出的自稳定性,以避免在操作过程中可能出现的故障，并在异常情况下,能够有良好的自我恢复机制。软硬件可靠性之间本质区别是后者为衰变和老化的物理机制，而前者是由于设计和执行错误。因此，软件的可靠性，在生产和测试阶段去考虑，已经太晚，必须在设计阶段确定。

**易用性**

设计应考虑到用户的实际操作现场，给用户的使用习惯充分考虑，如大多数人是手机双手拿或单手持握，单手时是右手还是左手来使用，操作哪个手指可以进行操作。尽可能多的人体工程学的应用，同时，设计时考虑到用户的使用习惯可以帮助避免手指触摸盲区。

**标准化原则**

标准化的好处很多，利用标准化可以最大限度的利用外部资源，可以和国际接轨，扩大软件的适用范围。

**实用性**

任何软件的存在的终极意义都是给用户带来便利，脱离了用户的软件都是不能长久的。

**可扩展性**

软件设计过程中应该使软件具有良好的扩展性，以便进行升级。

## 4.2 系统总体功能结构

### 4.2.1 功能框架

客户端是指某互联网金融企业平台的投资客户使用的产品集合体，用户可以通过应用软件来完成注册登录、项目查询、一键购买、订单管理、联系理财师和应用软件设置功能。一期的实现按照流程可分解为包括注册与登录流程、项目展示流程、购买与订单管理流程、收益查看流程、支付与提现流程、应用软件设置流程等。

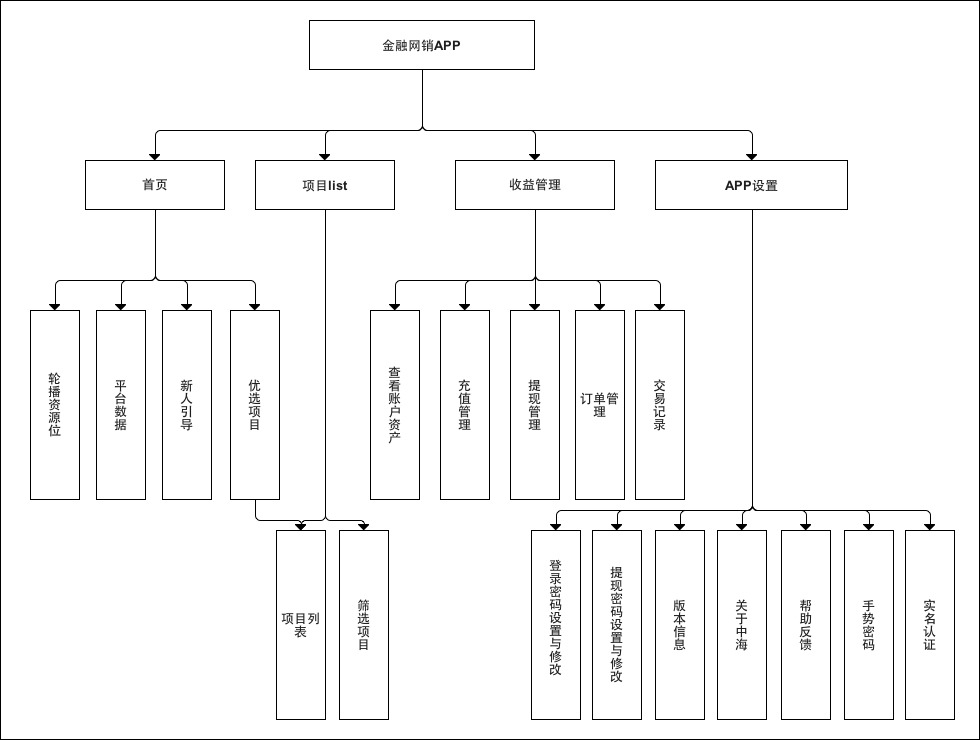


图4. 1 应用软件功能框架图

理财师方面，经过多轮沟通，他们希望通过我们可以打造一款销售工具，在这个工具中，可以协助他们完成资料的记录、工作方法的累积，让理财师可以快速的进阶。从而达成技术提高效率的目的。经过分析讨论，理财师管理后台包含我的工作台、客户管理、销售管理、员工管理、权限管理、理财师学院以及消息模块。

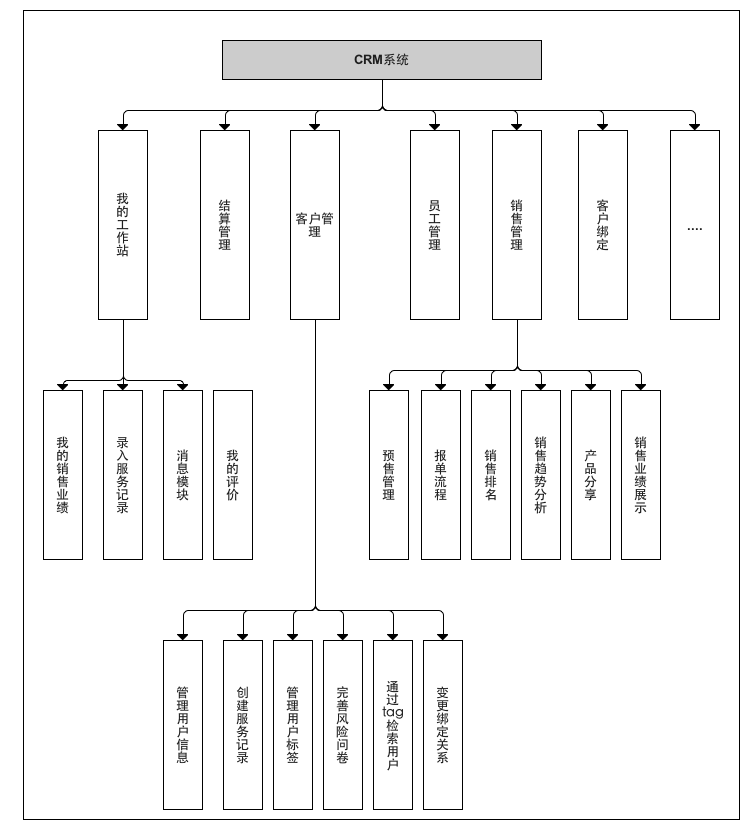


图4.2 理财师门店功能框架图

### 4.2.2 网络拓扑结构

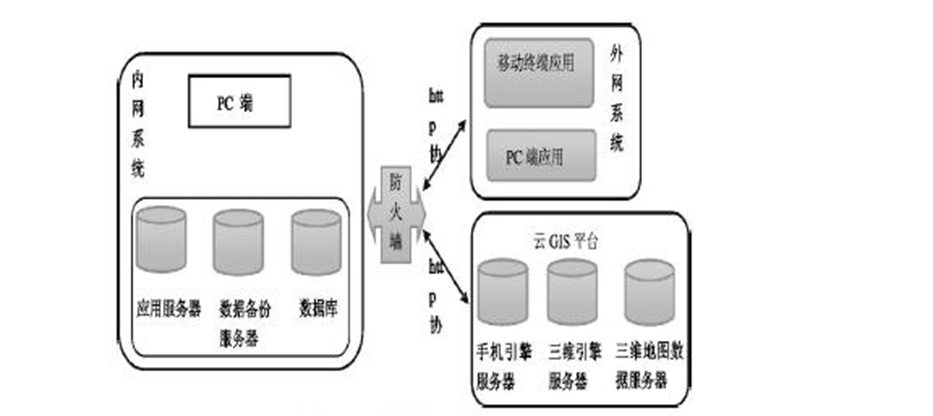


图4. 3 网络拓扑图

系统采用阿里云服务，阿里云作为国内最大的云服务提供商，具有强大的技术背景，很多对数据敏感的金融渠道商都将数据库建立在了阿里云。为了多重保障，平台建立了线下备份机，定时对线上数据库进行备份。线下数据库据有Linux防火墙，Linux防火墙稳定、性能可靠，可以及时的发现和解决问题，是备份数据库的最后的一道安全屏障。

## 4.3 系统技术实现架构

金融网络销售系统采用MVC模式开发，实现业务逻辑，数据，界面显示组织机构代码分离的MVC方法，部件起到了容纳业务规则的作用，并在UI需要个性化定制时，业务规则可以调用之前相同功能的模块。分层的设计理念，将软件分为视图，模型和控制器三个层次，每个层次都用来实现一定的功能[44]。View负责接收用户操作，数据输入和输出，Model主要用于处理数据请求和相应的业务逻辑来处理，Control收分发数据时，用户触发事件，处理不同的模型后，在到控制器不同的数据中选择对应的视图来显示数据。

本系统应用springMVC思想将应用软件实现框架设计为三个部分：ui部分（视图层）、业务逻辑部分（视图层）、业务逻辑部分（控制层）和数据存储部分（模型层）[45]，如下图所示：

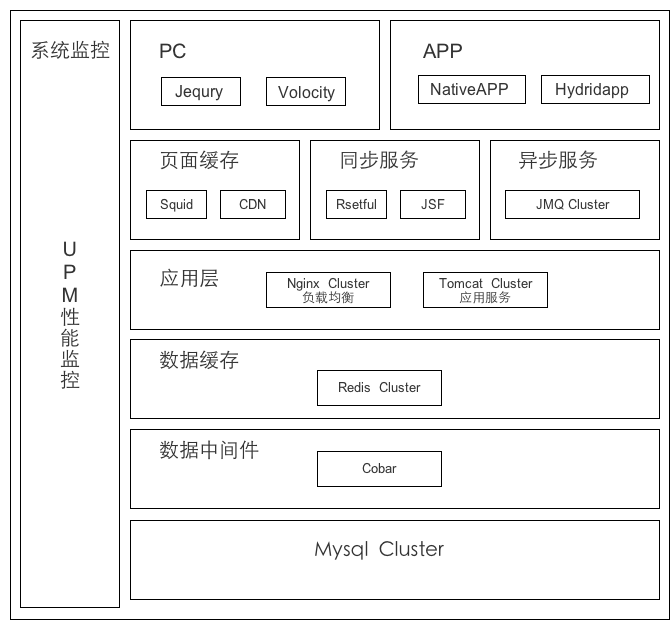


图4. 4 金融网络销售系统技术实现架构图

界面部分，即UI主要使用Velocity模板技术进行前端界面开发，前端工程师将静态页面完成，后台工程师按照MVC的框架开发整个平台功能，前端工程师只要要关注页面的ui效果，不需要写后端逻辑，当然前端和后端需要联调API，保证所有的借口都是可用的。后台工程师将逻辑实现并保存到数据库。Volicity模板是连接Model和view之间的链条，可以实现很多Java代码的功能，如if{}else{}，迭代循环等，大大丰富了前端开发的选择。

数据存储包含了各个业务模块抽象后的对象，使用了Cobar来对数据进行分库分表的存储，平台的数据都是通过调用服务器的API来说获取，并在本地BD和对一些用户信息和图片进行缓存。

业务逻辑部分，控制层在Spring框架下Controller的相关类为主，负责进行业务逻辑的处理。处理逻辑的时候，会根据需要调用相关的service后台服务，并通过JSF分布式框架、以Json格式异步访问数据库，修改或拿到相关数据。拿到数据后，在控制层进行数据解析，然后将解析后数据进行封装，根据也许的需求返回给视图层、或者是缓存到Redis集群。

## 4.4 数据库设计

金融类项目的字段比较多，某互联网金融企业应用软件也有庞大的数据表，限于篇幅，本文只列注册登录模块和购买支付模块的数据库设计。

。

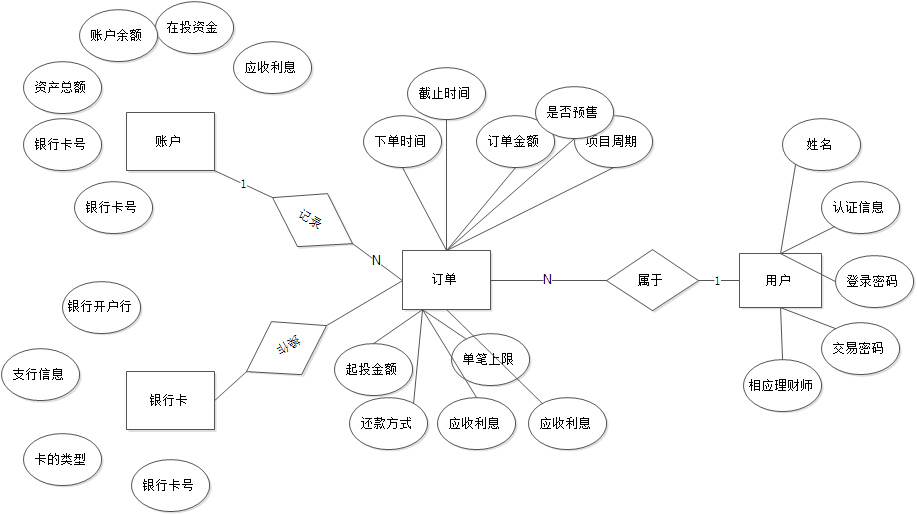


图4. 5 购买下单的模块的E-R图如如所示

1、deal(项目)

项目表是用来展示上单字段的一张表，包含了项目类型、项目来源、标题、副标题、项目icon、预期年化收益率、预售预期收益率、起投金额、投资单位、投资期限、投资返点、还款方式、总金额、已募金额、预售金额、已预售金额、投资进度、是否预售、是否推荐项目、项目当前状态、项目当前进度、创建时间开始公募时间等。这张表是是所有表中最重要的一张，几乎所有的业务都与这张表发生联系，如提现、还款、交易记录、在投资金等。

表4. 1 项目上单字段

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | 项目id |
| category | int | 项目类型 |  | 1:P2P 2:供应链 3:基金 4:信托 5:银行理财 6:资管 0:其他 |
| resource | string | 项目来源 |  |  |
| title | string | 标题 |  |  |
| subtitle | string | 副标题 |  |  |
| iconurl | text | 项目icon |  |  |
| apr | decimal | 预期年化收益率 |  |  |
| presale\_apr | decimal | 预售预期收益率 |  | 预期项目收益率=预期年化收益率+预售预期收益率 |
| base\_amount | decimal | 起投金额 |  |  |
| unit\_amount | decimal | 投资单位 |  |  |
| period | int | 投资期限 |  | 单位:天 |
| rebate | decimal | 投资返点 |  |  |
| repayment | int | 还款方式 |  | 1:到期还本付息 2:每月付息到期还本 3:等额本息 |
| total\_amount | decimal | 总金额 |  |  |
| raised\_amount | decimal | 已募金额 |  |  |
| presale\_amount | decimal | 预售金额 |  |  |
| raised\_presale\_amount | decimal | 已预售金额 |  |  |
| percentage | decimal | 投资进度 |  | 投资进度=已募金额/总金额 |
| is\_presale | tinyint | 是否预售 |  | 1:非预售 2:是预售 |
| is\_recommend | tinyint | 是否推荐项目 |  | 1:非推荐项目 2:推荐项目 |
| status | int | 项目当前状态 |  | 1:未开始 2:预售中 3:销售中 4:未计息 5:计息中 6:还款中19:已售完 99:已结束 |
| state | int | 项目当前进度 |  | 1:未上线 32:可卖 64:已下线 |
| create\_time | datetime | 创建时间 |  |  |
| presale\_begin\_time | datetime | 预售开始时间 |  |  |
| presale\_end\_time | datetime | 预售结束时间 |  |  |
| sale\_begin\_time | datetime | 开始公募时间 |  |  |
| sale\_end\_time | datetime | 结束公募时间 |  |  |
| repay\_begin\_time | datetime | 开始计息时间 |  |  |
| repay\_end\_time | datetime | 结束计息时间 |  |  |
| end\_time | datetime | 项目结束时间 |  |  |
| label1 | string | 项目标签1 |  |  |
| label2 | string | 项目标签2 |  |  |
| label3 | string | 项目标签3 |  |  |
| modify\_time | datetime | 最后更新时间 |  |  |

2.zhonghai.deal\_interest\_rate

项目的还款有很多中方法，其中后本后息和前息后本都是一次还款，等额本息需要多次还款，这张表便是用来记录每次还款本金及利息所占比例的表。包含了项目id、开始时间、结算时间、结算周期、实际收益率、预售收益率、是否还本金等字段。

表4. 2 还款利率表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| dealid | int | 项目id | Y |  |
| start\_time | datetime | 开始时间 |  | 本次计息周期开始时间 |
| settlement\_time | datetime | 结算时间 |  | 产生结算的时间点。对于定期收益项目，  只在结算时间点增加条目。对于供应链融资项目，每天增加实际收益率。 |
| period | int | 结算周期 |  |  |
| interest\_rate | decimal | 实际收益率 |  | 收益率\*本金=利息 |
| presale\_interest\_rate | decimal | 预售收益率 |  | 仅对于预售订单有效 |
| repay\_base | tinyint | 是否还本金 |  | 0:不还 1:还 |
| repay\_base\_percent | decimal | 还本金比例 |  |  |
| status | tinyint | 结算状态 |  | 0:未结算，1:已结算 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| dealid | int | 项目id | Y |  |
| sourceid | int | 合作伙伴id |  | 关联供应链表 |
| source\_desc | text | 合作伙伴介绍 |  |  |
| source\_link | text | 合作方链接 |  |  |
| source\_logo | text | 合作方LOGO |  |  |
| desc | text | 产品介绍 |  |  |
| max\_amount | bigint | 投资限额 |  |  |
| accumulate\_max\_account | bigint | 个人累计投资额度 |  |  |
| repay\_date | text | 还款日 |  |  |
| protect\_method | text | 保障方式 |  |  |
| finance\_purpose | text | 融资用途 |  |  |
| user\_agreement | text | 用户协议 |  |  |
| buy\_rule | text | 购买规则 |  |  |

3.zhonghai.deal\_detail

表4. 3 项目详情页字段

4.zhonghai.best\_deal

表4. 4 推荐项目

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| dealid | int | 项目id |  |  |
| type | tinyint | 优选类别 |  | 1 优选商品, 2 P2P专区, 3 资管专区, 4 银行理财 |
| iconurl | text | 优选项目ICON |  |  |
| sequence | int | 序号 |  |  |
| is\_show | tinyint | 是否展示 |  | 1:展示 2:不展示 |

5.用户

用户表拆分为账号表、基础信息表、账户表等三张表，分别定义如下：

zhonghai.login

表4. 5 账号表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | 用户id |
| username | string | 用户名 | Y | username与id一一对应 |
| password | string | 验证串 |  | 密码经加密变换后得到，为了账户安全，此处不应存储密码原文 |
| create\_time | datetime | 创建时间 |  |  |
| modify\_time | datetime | 最后更新时间 |  |  |

6.zhonghai.user\_info

表4. 6 用户信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| userid | int | 用户id | Y | 本字段由账号表查到 |
| idcard | string | 身份证号 |  |  |
| mobile | string | 手机号码 |  |  |
| gender | tinyint | 性别 |  | 1:男 2:女 |
| poiid | int | 门店id |  | 绑定后不允许修改 |
| bdid | int | 理财顾问id |  | 每位用户对应唯一的理财顾问 |
| extra | text | 补充信息 |  | 由理财顾问填写 |
| role | int | 角色 |  | 1:普通用户 2:理财顾问 3:门店经理 4:管理员 |

7.zhonghai.user\_safe\_info

表4. 7 账户设置表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| userid | int | 用户id | Y |  |
| question | string | 密保问题 |  |  |
| answer | string | 密保答案 |  | md5生成签名 |
| contact\_name | string | 紧急联系人姓名 |  |  |
| contact\_mobile | string | 紧急联系人手机号码 |  |  |
| contact\_relati  on | string | 紧急联系人关系 |  |  |

8.zhonghai.account

表4. 8 银行卡信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| userid | int | 用户id | Y | 本字段由账号表查到 |
| cardno | string | 银行卡号 |  | 每位用户对应唯一的银行卡号 |
| capital | decimal | 资产总额 |  |  |
| money | decimal | 账户余额 |  |  |

9.门店

此项下需要创建门店表和城市表，定义如下：

zhonghai.city

表4. 9 门店表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | 城市id |
| cityname | string | 城市名 |  |  |

10.zhonghai.region

表4. 10 城市表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| province | string | 省/直辖市 |  |  |
| city | string | 市 |  |  |
| district | string | 区/县 |  |  |
| code | string | 区域代号 |  |  |

11.zhonghai.poi

表4. 11 专属门店信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | 门店id |
| poiname | string | 门店名 |  |  |
| bdmid | int | 门店经理id |  |  |
| lat | decimal | 纬度 |  |  |
| lng | decimal | 经度 |  |  |
| address | text | 地址 |  |  |

12.zhonghai.poi\_account

表4. 12 专属门店信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| poiid | int | 门店id | Y |  |
| cardno | string | 银行卡号 |  | 每家门店对应唯一的银行卡号 |
| capital | decimal | 资产总额 |  |  |

13.理财顾问(BD/BDM)，zhonghai.bd\_login

表4. 13 bd注册信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | BD id |
| username | string | 用户名 | Y | username与id一一对应 |
| password | string | 验证串 |  | 密码经加密变换后得到，为了账户安全，此处不应存储密码原文 |
| create\_time | datetime | 创建时间 |  |  |
| modify\_time | datetime | 最后更新时间 |  |  |

14.zhonghai.bd

表4. 14 理财师信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| bdid | int | BD id | Y | 本字段由BD账号表查到 |
| bdname | string | 真实姓名 |  |  |
| poiid | int | 门店id |  |  |
| mobile | string | 手机号码 |  |  |
| gender | tinyint | 性别 |  | 1:男 2:女 |
| iconurl | text | BD头像 |  |  |
| is\_active | tinyint | 是否在职 |  | 1:在职 2:离职 |
| role | int | 角色 |  | 1:普通用户 2:理财顾问 3:门店经理 4:管理员 |
| create\_time | datetime | 创建时间 |  |  |
| modify\_time | datetime | 最后更新时间 |  |  |

15.预售码

zhonghai.presale

表4. 15 理财师创建的订单表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | 预售码id |
| code | string | 预售码 | Y | 32位的随机字符串 |
| userid | int | 用户id |  |  |
| user\_mobile | String | 预售码绑定用户手机号 |  |  |
| dealid | int | 项目id |  |  |
| bdid | int | 理财顾问id |  |  |
| poiid | int | 门店id |  |  |
| amount | decimal | 预售金额 |  |  |
| create\_time | datetime | 创建时间 |  |  |
| expiry\_time | datetime | 失效时间 |  |  |
| status | int | 当前状态 |  | 1:未使用 2:已使用 3:已失效 |

17.评价

zhonghai.rating

表4. 16 评价信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y | 评价id |
| userid | int | 用户id |  |  |
| bdid | int | 理财顾问id |  |  |
| rating | int | 评分 |  | 1~5之间 |
| yearmo | int | 年月份 |  | 格式:YYYYmm 例:201410 用户每月可给出一个评分，可修改当月评分 |
| create\_time | datetime | 评价时间 |  |  |
| modify\_time | datetime | 最后修改时间 |  |  |

18.投资收益表

表4. 17 投资收益表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| userid | int | 用户id |  |  |
| dealid | int | 项目id |  |  |
| bdid | int | 销售id |  |  |
| poiid | int | 门店id |  |  |
| rateid | int | 收益率id |  |  |
| orderid | int | 订单id |  |  |
| settlement\_time | datetime | 收益结算时间 |  |  |
| action\_time | datetime | 结算操作时间 |  |  |
| amount | decimal | 投资金额 |  |  |
| interest | decimal | 收益金额 |  | 按日按项目拆分 |
| presale\_interest | decimal | 预售收益 |  | 预售返点收益 |

19.账户流水表

zhonghai.balance

表4. 18 账户流水表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| outno | string | 流水号 |  |  |
| type | tinyint | 操作类型 |  | 1:转入 2:转出 |
| action | tinyint | 操作详情 |  | 1:充值 2:提现 3:购买项目 4:项目结息 5:项目返本 |
| userid | int | 用户id |  |  |
| cardno | string | 银行卡号 |  |  |
| amount | decimal | 金额 |  |  |
| balance\_before | decimal | 操作前余额 |  |  |
| balence\_after | decimal | 操作后余额(可用余额) |  |  |
| create\_time | datetime | 创建时间 |  |  |

21、项目筛选条件表

zhonghai.deal\_select\_menu

表4. 20 项目筛选条件表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| category | int | 项目类型 |  | 1:P2P 2:供应链 3:基金 4:信托 5:银行理财 6:资管 0:其他 |
| selectkey | string | 筛选条件 |  |  |
| name | string | 条件名称 |  |  |
| type | string | 筛选类型 |  | equal/range |
| values | string | 取值范围 |  | json格式 |
|  |  |  |  |  |

22、poi收益表

zhonghai.poi\_profit

表4. 21 poi收益表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| poiid | int | 门店id |  |  |
| userid | int | 用户id |  |  |
| dealid | int | 项目id |  |  |
| bdid | int | BD id |  |  |
| rateid | int | 收益率id |  |  |
| orderid | int | 订单id |  |  |
| settlement\_time | datetime | 收益结算时间 |  |  |
| action\_time | datetime | 结算操作时间 |  |  |
| amount | decimal | 投资金额 |  |  |
| interest | decimal | 收益金额 |  | 按日按项目拆分 |

24、商家返点结算记录

表4. 23 商家返点结算记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| poiid | int | 商家id |  |  |
| userid | int | 用户id |  |  |
| dealid | int | 项目id |  |  |
| bdid | int | 投资顾问 |  |  |
| interest | decimal | 收益 |  |  |
| remark | varchar(128) | 备注 |  |  |
| create\_time | datetime | 创建时间 |  |  |

25、zhonghai.loan\_contract\_info

表4. 24 合同信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| contract\_number | string | 协议编号 |  |  |
| contract\_date | datetime | 签订日期 |  |  |
| borrower\_id | int | 借款人id |  |  |
| lender\_name | string | 债权转让人姓名 |  |  |
| lender\_idcard | string | 债权转让人身份证 |  |  |
| lender\_username | string | 债权转让人用户名 |  |  |
| lend\_amount | decimal | 受让债权金额 |  | 单位:人民币元 |
| lend\_period | int | 债券期限 |  | 单位:天 |
| lend\_repayment | decimal | 每期应收本息 |  | 单位:人民币元 |
| thirdparty | text | 丙方:担保公司 |  |  |
| borrower\_name | string | 借款人姓名 |  |  |
| borrower\_idcard | string | 借款人身份证号 |  |  |
| finance\_purpose | text | 借款用途 |  |  |
| apr | decimal | 借款年化利率 |  |  |
| borrow\_amount | decimal | 借款本金金额 |  | 单位:人民币元 |
| borrow\_repayment | decimal | 月偿还本息金额 |  | 单位:人民币元 |
| repay\_month | int | 还款分期月数 |  |  |
| repay\_type | string | 还款方式 |  |  |
| repay\_date | int | 每月还款日 |  |  |

当用户购买了理财产品后，需要生成一份合同，合同中会写出借款人和还款人的信息，应还的数据已经享有的法律权利和义务。

26、用户绑定表

4. 25 绑定信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 类型 | 描述 | 索引 | 备注 |
| id | int | 自增主键 | Y |  |
| userid | int | 用户 |  |  |
| poiid | int | 绑定商家id |  |  |
| bdid | int | 绑定bdid |  |  |
| create\_time | datetime | 绑定时间 |  |  |
| modify\_time | datetime | 解绑时间 |  |  |
| status | tinyint | 绑定当前状态 0:无效绑定 1:现在绑定 |  |  |
| operatebd | int | 修改者的bdid |  | 注册时默认0 |

## 4.5 本章小结

本章主要讲了系统设计时参照的原则，以及系统的架构设计方案，然后从完整的业务组成上分析了金融网络销售的用户功能框架和门店管理用户框架，最后讲了数据库的设计以及产出的一些主要的表。

# 第五章 金融网络销售系统的详细设计与实现

## 5.1 金融网络销售用户端应用软件

### 5.1.1 注册模块

用户注册分为三步：

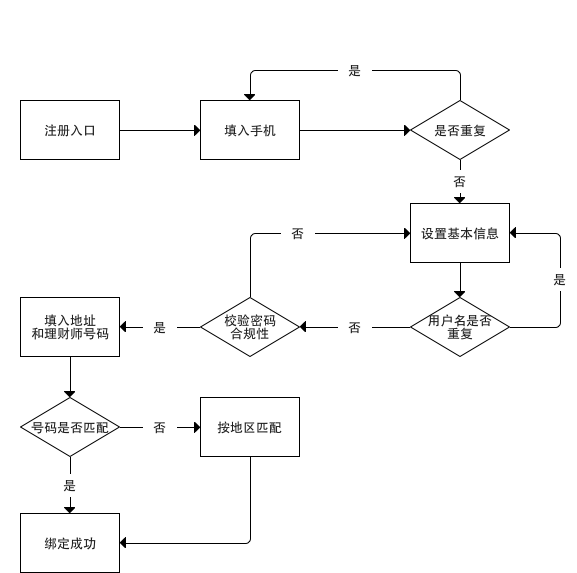


图5. 1 注册流程图

第一步需要填入手机号码。点击获取短信验证码，校验手机号码位数以及查重手机号码是否重复，每个手机号码只能注册一次，所有条件满足则触发短信平台，发送短信验证码，验证码显示为60s发一次，60s后可以重发,按钮上显示时间；填入验证码后点击下一步，正确的话进入下一步。

第二步设置用户名和登录密码。用户应该满足一定的规则，不能太简单或无限制，点击显示密码按钮，在输入框右边，点击显示密码；点击下一步按钮，点击下一步校验用户名重复性（最好可以做成异步）；点击右上角返回按钮，点击返回注册第一步。

第三步填入地址以及手机号匹配理财师。本页面会录入用户地址，地址是内嵌到本地，用户可链级选择，包含省市区选择控件，用滚轮控件选择；理财师手机号是选填手机号码是选填。

如图，理财师如果没有填写，理财师匹配由后台完成，逻辑如下：

a.填写地址和手机号区域冲突，但手机号为平台已注册理财师，以手机号为准

b.只填写地址，按现有系统逻辑匹配理财师

键入信息后点击提交后台进行校验，进入到完成页。在完成页可以选择直接进行实名认证页面或跳转首页。部分代码如下：

/\*\*

     \* 注册绑定理财师

     \* @param zhonghaiCode 地区码

     \* @param bdMobile     理财师手机号

     \* @param bdUsername   理财师姓名

     \* @throws BaseException

     \*/

    public void bindBd(int userId, String zhonghaiCode, String bdMobile, String bdUsername) throws BaseException {

        logger.info(String.format("【SERVICE START】绑定理财师.userId:%s, zhonghaiCode:%s, bdMobile:%s, bdUsername:%s",

                userId, zhonghaiCode, bdMobile, bdUsername));

        // 优先按手机号查询理财师

        Bd bd = bdDao.getByBdMoblie(bdMobile);

        if (bd != null) {

            // 绑定理财师

            UserInfo userInfo = userInfoDao.getByUserId(userId);

            userInfo.setBdId(bd.getBdId());

            userInfo.setPoiId(bd.getPoiId());

            userInfoDao.update(userInfo);

            UserBdRelation userBdRelation = userBdRelationDao.getByUserid(userId);

            userBdRelation.setBdId(bd.getBdId());

            userBdRelation.setPoiId(bd.getPoiId());

            userBdRelationDao.updateRelation(userId, bd);

            // 按用户名、地区查询理财师

        } else {

            bd = new Bd();

            Integer poiid = null;

            Integer bdid = null;

            // 按用户名查询理财师

            if (StringUtils.isNotEmpty(bdUsername)) {

                BdLogin bdLogin = bdLoginDao.getByUsername(bdUsername);

                if (bdLogin != null) {

                    Bd tmp = bdDao.getByBdId(bdLogin.getId());

                    if (tmp != null && tmp.getIsActive() == BdStatus.ACTIVE.getIntVal()) {

                        bdid = bdLogin.getId();

                    }

                }

            }

            // 根据地区码查找

            if (bdid == null && StringUtils.isNotEmpty(zhonghaiCode)) {

                Region region = regionDao.getByZhonghaiCode(zhonghaiCode);

                if (region != null) {

RegionPoiRelation relation =

regionPoiRelationDao.getByRegionId(region.getId());

                    if (relation != null) {

                        poiid = relation.getPoiId();

                        Poi poi = poiDao.getById(poiid);

                        if (poi != null) {

                            bdid = poi.getBdmId();  //有poi,bd就默认选bdmid

                            poiid = poi.getId();

                        } else {

logger.warn("【SERVICE WARN】指定门店不存在.poiId:" +

poiid);

                        }

                    } else {

                        logger.warn("【SERVICE WARN】该地区不存在门店.地区码:"

+ zhonghaiCode);

                    }

                } else {

logger.warn("【SERVICE WARN】不存在该地区.地区码:"

+zhonghaiCode);

                }

            }

            if (bdid == null) {

                bdid = Bd.DEFAULT\_BDID;//默认到总部理财师

                logger.warn("【SERVICE INFO】默认到总部理财师.bdid:" + bdid);

            }

            if (poiid == null) {

                // 省市区填写错误

                logger.warn("【SERVICE WARN】省市区填写错误");

                throw new BaseException

(ResultCodeConstants.BIND\_BD\_WRONG\_REGION);

            }

            bd.setBdId(bdid);

            bd.setPoiId(poiid);

            UserInfo userInfo = userInfoDao.getByUserId(userId);

            userInfo.setBdId(bd.getBdId());

            userInfo.setPoiId(bd.getPoiId());

            userInfoDao.update(userInfo);

            UserBdRelation userBdRelation = userBdRelationDao.getByUserid(userId);

            userBdRelation.setBdId(bd.getBdId());

            userBdRelation.setPoiId(bd.getPoiId());

            userBdRelationDao.updateRelation(userId, bd);

        }

        logger.info("【SERVICE END】绑定理财师.userId:" + userId);

    }

### 5.1.2 登录模块

登录页包含：用户名、密码、忘码入口、注册入口。

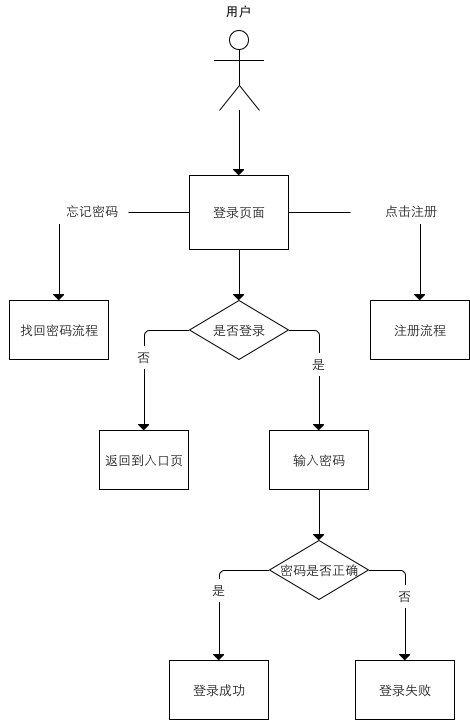


图5. 2 登录流程

进入登录页面后，按照页面提示输入相关信息（name和password），提交进行登录操作，错误时进行提示；用户如果忘记password，运用找回密码的功能，按照流程就可以重新设置密码；对于新的用户，则会别引导去注册流程。主要代码如下图：

public Login login(String username, String rawPwd) throws BaseException {

String encrypt = MD5Util.generateMd5("++" + rawPwd + username);

Login login = loginDao.getByUsername(username);

if (login == null) {

throw new BaseException(ResultCodeConstants.USER\_NOT\_EXIST);

} else if (!encrypt.equals(login.getPassword())) {

throw new BaseException(ResultCodeConstants.LOGIN\_PWD\_WRONG);

}

throw new BaseException(500010);

}

### 5.1.3 项目展示模块

项目展示主要的功能是让用户进入应用软件后，由粗到细的看到项目，通过首页的精选项目和项目list页以项目详情页来完成。

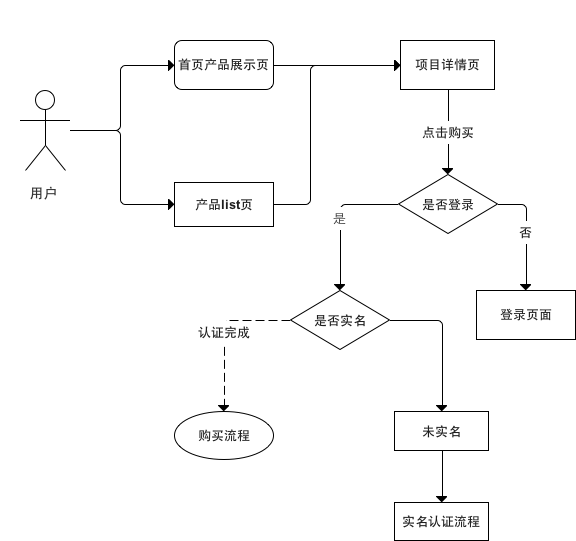


图5. 3 项目展示流程图

打开应用软件到达首页后，可以从首页的精品项目展示进入项目详情页，也可以从底部tab进入到项目list页，再由产品list页进入项目详情页。在项目详情页，通过立即购买按钮，这里需要区分用户的状态，已登录用户需要校验是否已经实名，已实名的用户可以直接到达订单页，未登录的用户则要去登录界面。

登录态分为已登陆和未登录：未登录态下点击“账户资产”跳转登录页面，点击“账户设置”展示部分页面。包含：关于我们、评价一下、更多；登录状态下首页有一个快捷菜单按钮，点击展开 list，包含：充值、我的订单、理财师、帮助中心。页面尾部包含下拉去除遮挡交互，遮挡部分包含新人引导和安全保障。页面元素包含：轮播banner，数量暂不设上限，由后台控制，尺寸见设计图，未登录依然可跳到相关页面；快捷菜单按钮，点击展开 list，包含：充值、我的订单、理财师、帮助中心，点击可跳到相应页面；精品项目展示，项目展示包含项目类型、进度条、年化收益率、起投金额、项目总额、理财期限。

产品list页

产品list页一期只做项目卡片展示，二期及以后考虑加入筛选等其他功能。无论项目是否在售，点击卡片都跳转项目详情页。项目卡片元素包含：项目名称，不超过13个字；项目状态按钮a.预售中b.去投资（立即购买）c.已售罄，计息时间还未到d.计息中e.已计息f.已还款g.已结 项；项目进度条；预期年化收益率，保留小数点后四位；项目期限，单位是天；项目总额，单位是元；还款方式，包含”先息后本“、”等额本息“、”到期还本息“；保障方式，一期固定不变

产品详情页

详情页的入口目前有三个：首页精选项目，产品list页，我的订单点击项目卡片。

页面包含元素：项目名称；年化收益率，；项目总额，单位是万元；剩余可购金额；项目周期，单位是天；基本信息包含a起投金额：随项目发生变化，b.还款方式包含“先息后本”、“等额本息”、 “到期还本息”,c.合作方d.保障方式：一期是固定，二期及以后会考虑随项目改动 ；项目介绍；投资记录；购买记录；购买按钮,交互：按钮悬停在页面下方，滑动页面时消失，停止滑动时出现,包含状态:a.预售中，b.去投资（立即购买），c.已售罄，计息时间还未到，d.计息中，e.已计息，f.已还款，g.已结项。

投资记录

每页默认展示20条，每条投资记录包含三个字段1）投资人：只露出用户名第一个和最后一个字符，其余星号处理；2）投资金额：默认展示到元；3）投资时间：具体到分，分两行展示。

### 5.1.4 购买与订单管理流程

购买和订单管理流程是应用软件的核心流程包含了项目详情页、订单页、支付反馈页、订单管理页以及交易记录页面。

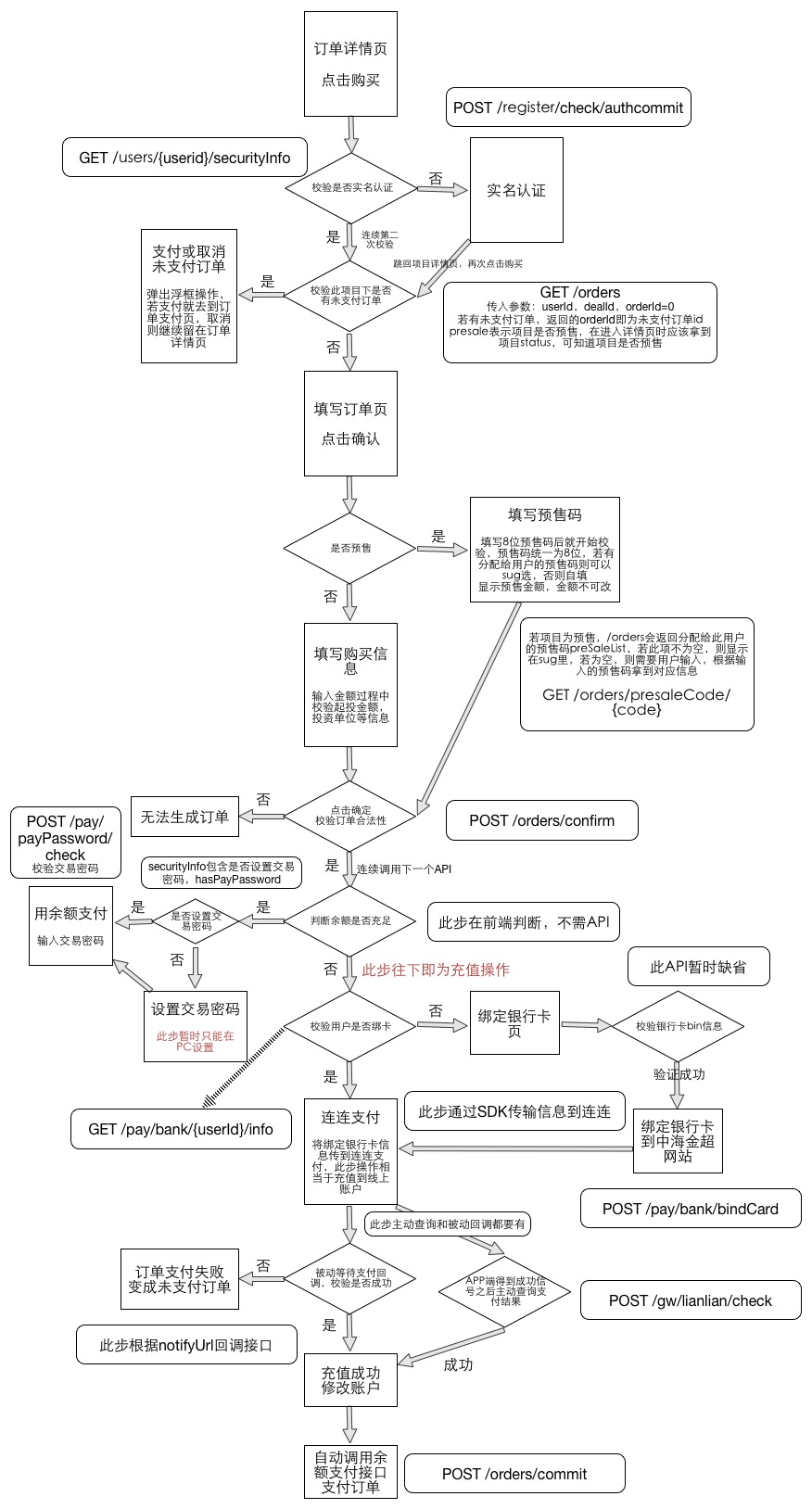


图5. 4 购买下单流程图

购买项目的入口目前有三个：首页精选项目，产品list页，我的订单点击项目卡片。

产品详情包含了项目信息以及以往的交易记录。页页面包含元素：项目名称；年化收益率，；项目总额，单位是万元；剩余可购金额；项目周期，单位是天；基本信息包含a起投金额：随项目发生变化，b.还款方式包含“先息后本”、“等额本息”、 “到期还本息”,c.合作方d.保障方式：一期是固定，二期及以后会考虑随项目改动 ；项目介绍；投资记录；购买记录；购买按钮。

购买按钮包含项目状态，不同的状态展示不同的文案，包含:a.预售中，b.去投资（立即购买），c.已售罄，计息时间还未到，d.计息中，e.已计息，f.已还款，g.已结项。

由于无论余额是否充足，支付流程的最后一步都是余额支付，所以此步骤需要进行多一步的校验，解决方案是加一个UUID，输入交易密码和通过第三方充值成功，二者任取其一都会生成一个UUID存在memcache中，校验方式类似于短信验证码，把这个UUID传给最后一步的API，校验成功后就用余额支付。

在校验支付密码的API最后要生成UUID，同样修改第三方支付的回调接口和主动查询接口，最后都要生成UUID并返回。最后支付成功。

部分实现代码如下：

/\*\*

\* 生成订单

\*

\* @param presaleCode 预售码

\* @param dealId 项目id

\* @param amount 金额

\* @param bdid 理财师id

\* @param poiid 门店id

\* @return

\* @throws com.weejinfu.common.exception.BaseException

\*/

@Transactional(value = "transactionManager", rollbackFor = Exception.class)

public int resolveOrder(String presaleCode, int dealId, BigDecimal amount, int userId, int bdid, int poiid) throws BaseException {

//同一个deal,如果有未支付的订单就直接返回该订单id

int orderId = getUnpayOrderForDealid(dealId, userId);

if (orderId > -1) {

return orderId;

}

Deal deal = dealService.getDealById(dealId);

//判断deal的状态

checkDealStatus(presaleCode, deal);

//判断订单的额度

checkOrderAmout(amount, deal);

Presale presale = null;

// 若为预售项目，则检验预售码

if (deal.getStatus() == DealStatus.PRESALING.getIntVal()) {

if (StringUtils.isNotEmpty(presaleCode)) {

// 验证预售码是否有效；同时锁定预售码

presale = presaleService.verifyCode(presaleCode, dealId, amount,

userId, true);

} else {

throw new BaseException

(ResultCodeConstants.ORDER\_PRESALE\_CODE\_ERROR);

}

}

//库存池消减；锁定金额，倒计时（过期释放金额，修改订单状态）；

日志

BigDecimal availableAmount =

deal.getTotalAmount().subtract(deal.getRaisedAmount());

if (amount.compareTo(availableAmount) == 1) {

// 项目可购额度不足订单额度

throw new BaseException(ResultCodeConstants.

ORDER\_DEAL\_LEFL\_AMOUNT\_NOT\_ENOUGH);

}

ZhOrder order = new ZhOrder();

order.setUserId(userId);

order.setDealId(dealId);

if (bdid > 0) {

order.setBdId(bdid);

order.setPoiId(poiid);

}

order.setOrderTime(new Date());

order.setAmount(amount);

order.setStatus(ZhOrderStatus.UNPAY.getCode());

if (presale != null) {

order.setIsPresale(Deal.PRESALE);

order.setPresaleId(presale.getId());

} else {

order.setIsPresale(Deal.COMMON\_SALE);

}

//插入订单

orderId = zhOrderDao.insert(order);

//更新预售码状态为已使用

if (presale != null) {

presale.setStatus(Presale.STATUS\_USED);

presale.setUserid(userId);

presaleService.update(presale);

// 更新项目预售标满时间

dealService.updatePreSaleReadyTime(dealId);

}

return orderId;

}

### 5.1.5 收益资产查看流程

该流程一期以展示为主，具体页面包含：我的资产页、账户资产页、我的订单、交易记录、理财师页面、提现银行卡、充值页面和提现页面。

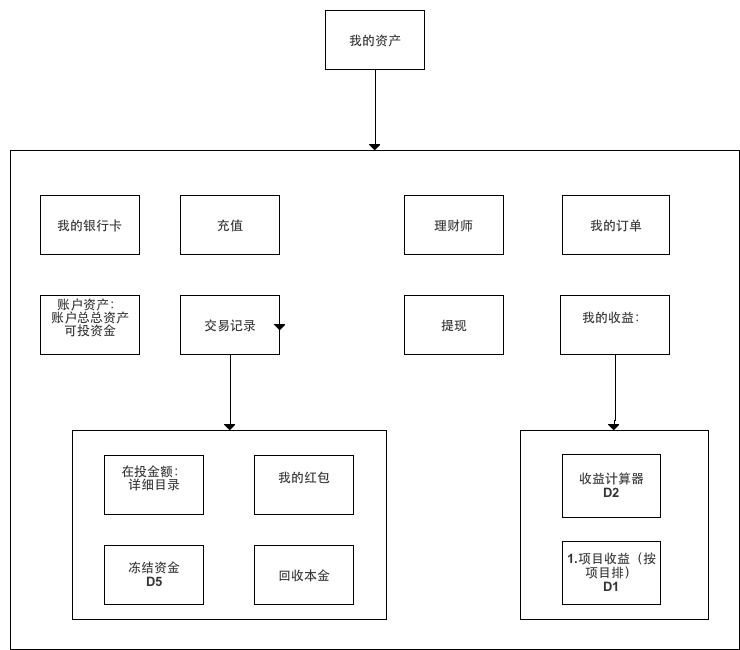


图5. 5 收益资产查看流程

我的资产

我的资产分为登陆态和未登录态。登陆状态下，点击跳转资产页；未登录状态下点击logo，点击跳转登陆页面。页面元素包含：昨日收益，由后端传送；累计收益，已到账的收益的累和；未到账收益，处于已支付但未到账的收益；可用余额，项目的有效充值和已到账收益中除去提现手续费，冻结资金及在投资近所剩下的金额；5）账户资产，点击可到二级页面；我的订单，点击可管理订单，查看合同，取消订单，继续支付等；交易记 录，可以查看收益，充值，提现进度，手续费等记录；理财师，可查看理财师；提现银行卡，一期只做对银行卡的展示；充值，这个页面的情况有待确定；提现，如果已经绑定了提现银行卡，可以发起提现流程。

账户资产

页面元素包含：账户总资产，等于在投资金+可用余额+冻结资金，保留两位小数点，单位为万元；可用余额，项目的有效充值和已到账收益中除去提现手续费，冻结资金及在投资近所剩下的金额；在投金额，目前出于已付款和已结项之间的投资累计；冻结资金，提现申请的金额；累计投资，从注册开始到现在；未到账收益，包含等额本息中未到账的利息、到期本息中的未到账利息、先息后本方式中暂未还款的利息

我的订单

订单页主要完成对项目基本信息的确认以及对订单信息的录入。可以从项目详情的立即购买和未完成订单进入订单页。页面元素包含：1）项目名称；2）订单金额前端校验：不能是负数，后端校验：a.是否超过单笔限额、b.是否是超过剩余可购、c.是否小于单笔限额；还款方式包含三种：a.等额本息、b.先本后息、c.到期本息；还款日期；支付方式（这块还需要在细致~）包含两种方式：a.余额支付，余额大于订单金额时用余额支付、b.快捷支付，需要先绑定银行卡；确认按钮

交易记录

从资产页和提现页面的查看提现记录中可以进入交易记录。交易记录包含四个字段：交易类型、时间、金额、账户余额。交易记录中提供筛选功能，包含提现和充值，收益，提现手续费、额外津贴；交易时间具体到分钟，在类型下显示；金额具体到小数点后两位；账户余额精确到小数点后两位。

我的理财师

目前一期理财师的功能会比较简单，单纯的做一个展示，带到二期会加入推荐好友，理财问卷、访谈预约、生成预售码。页面元素包含：1）头像2）姓名3）手机4）所属门店5）联系理财师button，点击button，会弹出拨号确认弹框。

提现银行卡

目前一期只支持一张卡的展示，不支持绑定和修改。2.如果没有绑定银行卡，后台判断：检测到未实名，弹框提示实名，点击确定跳转到实名认证页面，点击取消回到“我的资产”；检测到已实名，但未绑定银行卡，提示去pc绑定银行卡，文案：您还未绑定银行卡，为安全起见，请去电脑上进行绑定！页面元素包含：银行logo、银行名称、4位尾号。

充值

检测用户是否实名，未实名：弹框提示实名，点击确定跳转实名认证页面，点击取消跳转“我的资产”页面；已实名：直接点击去进行充值。

提现页面

点击提现按钮跳转提现页面，目前只有这一个入口，提现的手续费是系统自动计算；如果未绑定提现银行卡，则弹出提示“请先在电脑上绑定提现银行卡”，确认弹框后，回到“我的资产”页。页面包含元素：持卡人，必须是实名认证的用户名；提现银行卡，展示尾号4位；可提现金额，有后端传送；提现金额前端校验：必须是大于0的正数，后端校验：如果超出账户余额，提示“提现金额必须小于账户余额！”；实到账金额,扣除手续费之后的金额；提现手续费：按照渠道的计算逻辑来定；到账时间提示文案：“提现将在两个工作日内到达银行卡，具体时间视不同银行而定”。

### 5.1.6 应用软件设置流程

用户可以在设置流程里里完成安全方面的设置，也可以查看一些文本性的展示内容，如关于中海、版本信息、用户协议等。一期账户设置包含了实名认证、登录密码、提现密码、密保问题、关于中海、帮助中心、版本信息、检查更新等。其中提现密码及密保问题的设置都放在了二期去做，一期只做引导。

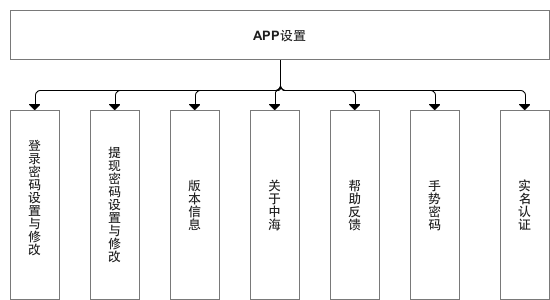


图5. 6 设置功能模块图

账户设置页面

账户设置页面作为设置的入口，承载了安全中心各项功能，点击都会进入到相关的二级页面，账户设置包含了安全设置和展示信息的入口。页面元素包含：实名认证，点击进入实名认证页面；登录密码包含3个子页面a.验证旧的登陆密码、b.验证手机号码、c.重置登陆密码；提现密码，只做展示；关于我们；帮助中心；支持一下我们；8）更多；退出登陆。

登陆密码

重置登陆密码分两步完成：第一步是验证旧的密码，如果忘记旧密码可以通过“忘记密码”找回密码；第二步是重置登陆密码。页面元素：旧密码输入框、忘记密码、提交按钮。忘记登陆密码分两步完成：第一步是验证手机号，手机号正确进入下一步；第二步是重置登陆密码，页面元素：显示绑定手机号，中间四位用星号代替；获取验证码按钮，点击后上面显示下次可以 获取的时间（56s）；验证按钮。忘记密码和重置登陆密码通过了第一步之后都会进入重置登陆密码页面元素：新密码输入框，框内引导语：请输入新的密码；框的右侧由一个显示密码的按钮，点击显示密码；确认按钮。

手势密码

用户下载应用软件注册完成，登陆后弹出设置手势密码页面，可以跳过，跳过后，默认关闭手势密码。下次登陆不再提示手势密码。如果设置，则手势密码为打开。账户设置的点进手势密码设置页面的功能包含：开关、显示轨迹开关、修改手势密码、忘记手势密码（验证手势密码页面展示）和用其他账号登陆（验证手势密码页面展示。页面包含：

* 开关：用户下载完应用软件后如果设置了手势密码则开关开启；点击关闭，校验旧的手势密码，校验通过，手势密码关闭；忘记了手势密码，则可以点击“忘记手势密码”，跳转到验证登陆密码页面，验证通过后直接设置新的手势密码。用其他账号登陆只在验证手势密码页面展示，点击后退出登陆态，展示登陆页面（填写用户名和密码）。
* 显示轨迹开关，可以直接关闭和开启，不需要验证。
* 修改手势密码，步骤如下：设置新的手势密码、确认新的手势密码。
* 忘记手势密码：跳转校验登陆密码页面，不显示用户名，只校验登陆密码，校验通过后重新设置手势密码，校验不通过及退出登陆态。

实名认证

实名认证作为平台业务的前置条件,在如下操作里都要进行check：绑定银行卡、充值、下单、设置交易密码、设置密保问题。如果检查到用户未实名，弹框提示用户去实名。页面元素包含姓名填框、输入身份证、点击提交按钮。实名认证调用的是第三方的API，用户每次点击调用一次。实现代码如下：

/\*\*

\* 实名认证

\*

\* @param userId 用户id

\* @param identityNo 身份证号

\* @param realName 真实姓名

\*/

public void authentication(int userId, String identityNo, String realName) {

if (StringUtils.isBlank(identityNo)) {

throw new BaseException

(ResultCodeConstants.AUTH\_IDNUM\_IS\_EMPTY);

}

if (StringUtils.isNotBlank

(IDCardUtil.IDCardValidate(StringUtils.lowerCase(identityNo)))) {

throw new BaseException

(ResultCodeConstants.AUTH\_IDNUM\_INLLIGA);

}

if (StringUtils.isBlank(realName)) {

throw new BaseException

(ResultCodeConstants.AUTH\_NAME\_IS\_EMPTY);

}

UserInfo u = userInfoDao.getByIdentityCard(identityNo);

if (u != null) {

throw new BaseException

(ResultCodeConstants.AUTH\_WAS\_AUTHED);

}

Account account = payAccountDao.getAccountByUserId(userId);

if (account == null) {

throw new BaseException

(ResultCodeConstants.AUTH\_HAVE\_NO\_ACCOUNT);

}

// 用户未认证过

if (StringUtils.isBlank(account.getIdentityCard())

|| StringUtils.isBlank(account.getUserName())) {

boolean success = AuthUtil.validator(realName, identityNo);

//认证成功入库

if (success) {

account.setUserName(realName);

account.setIdentityCard(identityNo);

account.setVerifyNameTime(new Date());

payAccountDao.updateAuthenticationInfo(account);

// 添加用户性别

UserInfo userInfo = userInfoDao.getByUserId(userId);

userInfo.setGender(IDCardUtil.parseGender(identityNo));

userInfo.setRealName(realName);

userInfo.setIdentityCard(identityNo);

// 判断性别

if (null != identityNo && !identityNo.isEmpty()) {

String day = identityNo.substring(6, 14);

int gender = Integer.parseInt

(String.valueOf(identityNo.charAt(16)));

if (gender % 2 == 0) {

userInfo.setGender(2);

} else {

userInfo.setGender(1);

}

userInfo.setBirthday(Integer.parseInt(day));

}

userInfoDao.update(userInfo);

} else {

throw new BaseException(ResultCodeConstants.AUTH\_FAILED);

}

} else {

if (!identityNo.equals(account.getIdentityCard())

|| !realName.equals(account.getUserName())) {

throw new BaseException(ResultCodeConstants.AUTH\_FAILED);

}

}

}

### 5.1.7 交易密码

应用场景：

**（1）更换绑定手机号**

首先校验实名认证以及交易密码设置：1)未实名（未实名一定未设置交易密码）,点击绑定手机的"更换"，显示文案:修改绑定手机需要完成实名认证和交易密码设置!Button:去实名,点击跳转实名认证页面;取消回到当前页面，完成实名认证,回到安全设置页,实名状态更新为完成。2)已实名但未完成交易密码,进入安全设置列表.绑定手机列表。点击“更换”，显示文案：您需要先设置交易密码,设置成功后可修改绑定手机号码。Button：“去设置”点击跳转 交易密码设置页面；取消回到当前页面。3）已实名且交易密码已设置,点击进入设置流程:验证交易密码；验证手机短信验证码；显示已绑定手机号,隐藏中间四位验证码有效时长10分钟,输错后未超时都可以再用；验证新手机，后端校验是否是已绑定手机号，提示文案“该手机号码已存在，请重新输入”点击"确定",会到绑定页面；修改成功，在设置成功页,放置"返回安全设置",点击返回安全设置列表。

实现细节：

此处需要加一个验证交易密码的接口：POST /users/{userId}/tradePassword/check (mobile端接口不要带userId，通过token获得)

**（2）全额余额交易**

在项目详情页点击"购买"

1)校验到未实名且未设置交易密码，文案:为了您的账户安全,购买项目需要完成实名认证和交易密码置!Button:去设置,点击跳转实名认证界面;取消,返回项目详情页。

2）校验已实名,但未设置交易密码.弹框提示，文案:您需要先设置交易密码,设置成功后可进行项目购买；Button:去设置,跳转交易密码设置页;取消,回到项目详情页，在交易密码设置完成页，显示“返回订单页”，点击进入订单页面。

3）余额充足时，需要输入交易密码才能进行支付。点击立即购买,弹出弹框，文案:将从您的账户中扣除XXXXXXXx元,请输入交易密码!点击确定跳转支付成功页,点击取消会到订单页面。

异常情况:订单金额大于预售可购额度时,提示错误，文案："抱歉！您的订单金额大于项目可购余额,请重新下单。点击“我知道了”，跳转订单页。

4）若有余额但余额不足时，通过银行卡充值然后余额支付，此时不再输入交易密码

实现细节：

由于无论余额是否充足，支付流程的最后一步都是余额支付，所以此步骤需要进行多一步的校验，暂定解决方案是加一个UUID，输入交易密码和通过第三方充值成功，二者任取其一都会生成一个UUID存在memcache中，校验方式类似于短信验证码，把这个UUID传给最后一步的API，校验成功后就用余额支付；在校验支付密码的API最后要生成UUID，同样修改第三方支付的回调接口和主动查询接口，最后都要生成UUID并返回。

**（3）申请提现**

输入提现金额，校验金额，输入交易密码，验证通过后即可提交申请，错误提示:您输入的交易密码错误,请重新输入!

异常情况：在充值提现页面,点击提现.

1）校验到未实名，弹框提示去实名（未实名一定未设置交易密码）

文案：为了您的资金安全,提现需要先完成实名认证、绑定银行卡和交易密码设置!Button:去设置,点击跳转实名认证界面;取消,返回充值提现页面

2）校验到未绑定银行卡，弹框提示绑卡，文案：为了您的资金安全,提现需要先完成绑定银行卡和交易密码设置!Button:去设置,点击跳转绑卡界面;取消,返回充值提现页面

3）校验到未设置交易密码,弹框引导去设置，文案：您需要先设置交易密码,设置成功后可进行提现操作。Button:去设置,点击跳转交易密码设置页面;取消,返回充值提现页面。

**（4）设置交易密码**

设置交易密码前需要先完成实名认证,认证完成后可以设置交易密码。

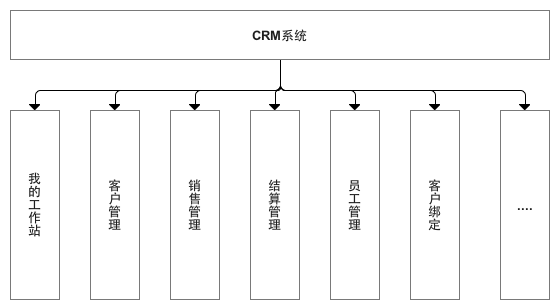
输入两次交易密码，校验通过即设置成功，密码为六位。

**（5）修改交易密码**

步骤：1）校验旧的交易密码2）输入新的/确认新的交易密码3）设置完成后,点击"返回安全设置",回到安全列表。

## 5.2 门店管理功能

门店管理是理财师用的平台，通过门店管理功能，理财师可以对自己的用户进行分类管理，可以对自己的销售业绩进行查看，同时，如果门店有多位理财师，还可以进行部门管理。主要包含的功能模块包含我的工作台、客户管理、销售管理、结算管理、员工管理、客户绑定及其它功能。



\

图5. 7 门店管理框架图

主要页面与功能

### 5.2.1 我的工作台

【我的工作台】页面基于bd后台的【业绩直达】页面修改。

主要功能点

1）顶部banner，用于对渠道的运营，可参考目前线上对和美家的示例。具体尺寸由设计师给出。

2)业绩速览，现有数据展示模块，不做修改，直接复用。

3)在线项目列表,基于现有【平台在线项目】模块,去掉“到期日”和“状态”2个字段,修改“门店销售金额”的单位为“万”，修改“预期收益”文案为“销售返佣”,视觉上缩短【在线项目】模块的宽度，一个模块最多展示10个在线项目，如果超出则翻页。

4）新增模块，展现BD收到的通知和消息（通知指官方群发消息），每一个条目统称为信息。点击more，当前页面跳转，进入信息列表页。点击标题，新tab打开信息的详情页。点击后未读标记消失。

BD可以收到的信息来源包括： 预售码 、学院消息、平台公告、工作站消息、用户消息、项目通知。

5）服务速记，新增模块，方便BD快速记录线下拜访行为。数据与【客户管理】-【客户详情】-【服务记录】中的数据，保持一致点击【记录】。按钮前，BD必须选择服务类型中的一种，默认为“主动拜访”。下拉展示：BD必须填写客户姓名



图5. 8 姓名输入框

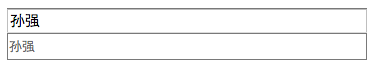
填写时，前端根据用户已经填写的文字预判目标姓名（sug提示功能），姓名从该BD下属用户中查找，例如一个成功的case

图5. 9 成功的例子

一个失败的例子

图5. 10 失败的例子

如果BD要填写【备注】这一类型的服务记录，则必须填写服务记录的文字部分，而其他服务类型可以不填写文字，直接生成服务记录。

6）我的服务记录

新增模块，展示BD本月内的服务记录情况；展示“服务计数”字段，累计该BD本月生成的服务记录的条数； 展示“服务客数”字段，累计、去重该BD本月服务客户的个数；展示“获得评价”字段，累计投资客户给该BD点击评价的次数，同时以条形图的方式，展示不同类型评价的比例

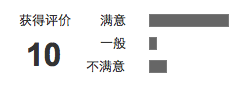


图5. 11 评价的比例

点击more，进入【客户管理】-【客户详情】-【服务记录】页面

7）客户风险评估·完成度



图5. 12 风险问卷完成度

3/36表示该BD名下36个客户中，有3个客户已经完成；“现在去完善”为按钮或链接，点击后，进入【客户管理】-带未完成问卷tag的筛选页。

### 5.2.2 客户管理

【客户管理】页面基于原BD后台的【客户管理】页面

主要功能点

1）数据速览

2）客户管理，客户列表默认按加入时间倒序展示（新->老），增加搜索模块，搜索框支持按客户姓名（如：小强）、手机号（如：13811111111）点击【检索】按钮，搜索结果展示在客户列表中。点击【标签筛选】按钮，下拉展示出tag列表



图5. 13 tag列表

每个tag都由“tag名+tag被该BD名下客户使用的人数”组成，例如男性对于投资偏好类的二级标签,采用"偏好|xx|2"的显示方法。偏好包括"曾投","偏好","抵触"，点击tag，tag呈现被点击态，同时客户列表展现按该tag的筛选结果。可以点击多个tag选取交集，如“男性”+“中年”+“谨慎性”。再次点击tag可以取消点击态。

【编辑标签】点击"编辑标签",标签进入可编辑态（需要视觉上区分）

按钮文案变为“退出编辑”,点击按钮即可退出编辑态；在可编辑态，点击标签，弹出标签详情弹窗编辑标签名,标签名a变为b,点击保存后校验:如果b不存在,则创建b,a的用户变更到b下面,清除a；如果b存在,则提示错误.文案:该标签已经存在,请勿重复添加。标签客户 20/36 表示：该标签被BD明显36个客户中的20个使用。



图5. 14 快速检索输入框

快速搜索和定位到该BD名下的某个客户，在筛选界面包含checkbox,用户信息.禁掉全选功能。包含确定和取消按钮，"确定"button,点击后将勾选结果带到用户选择页；"取消"button,点击后无论是否勾选checkbox,都回到用户选择页,保持此次筛选前的结果。标签标记客户明细，可以通过点击checkbox修改标签标记的名单。



图5. 15 修改标记的名单

退出，不保存直接关闭弹窗；保存，必须选取至少一个客户，否则保存按钮不可点，保存后关闭弹窗；删除标签，删除标签对客户的标记关系，但不影响客户资料；官方标签不可删除，只有BD自己创建的标签可以删除，相应的删除按钮需要设置为不可点。点击【创建新标签】弹出新建弹窗



图5. 16 新建标签弹框

点击客户列表的【查看详情】，进入【客户信息】页，用户信息页

图5. 17 用户信息页面

用户资料-地址，修改为“地区”，记录客户填写的地理位置信息；投资历史模块不变，去掉备注信息模块。分类展示该客户被赋予的tag； 当鼠标放在标签上时,显示删除按钮 :

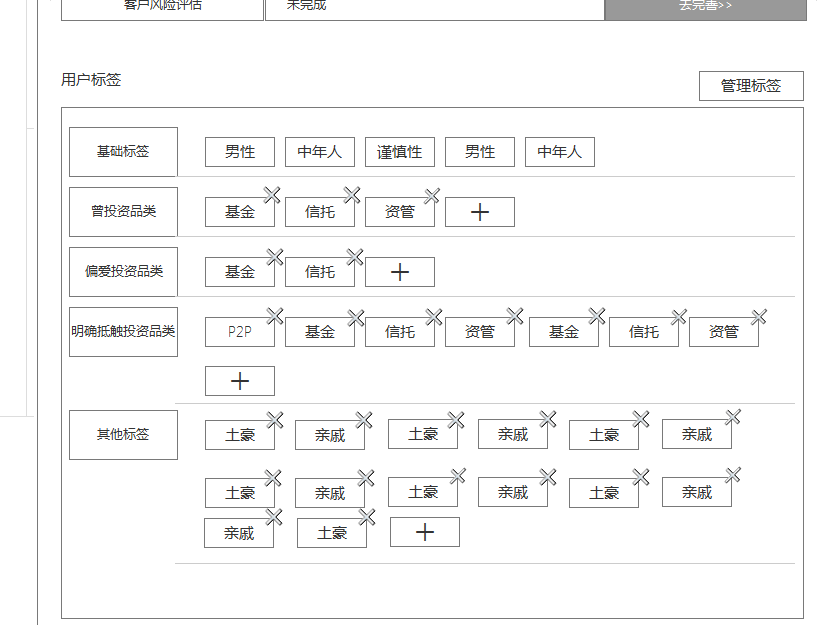


图5. 18 编辑用户标签

 BD可以在每个分类后，点击“+”符号，弹出添加标签页面.添加标签页面展示的标签list是用户尚未添加的标签，曾投资品类,偏爱投资品类,明确抵触投资品类展示该用户尚未使用的标签.

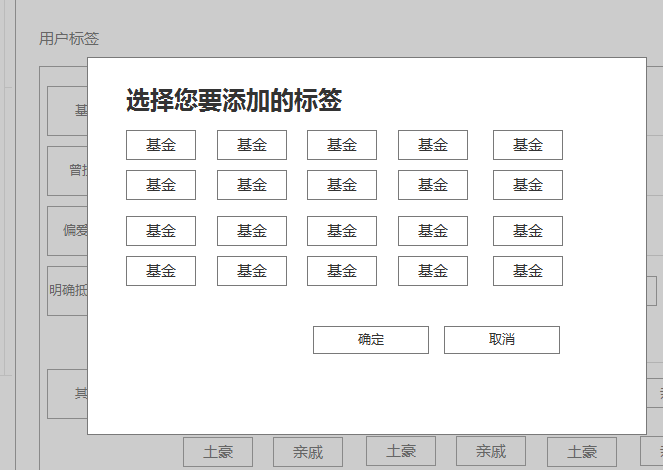


图5. 19 添加标签页面

在"其他标签"点击添加直接跳转到标签添加页面,默认勾选用户消息。

图5. 20 对单个用户添加标签

本页面没有快速筛选用户条框，用户名前的checkbox默认选中，点击保存,回到客户标签页面,保存选择结果。



图5. 21 客户标签页面

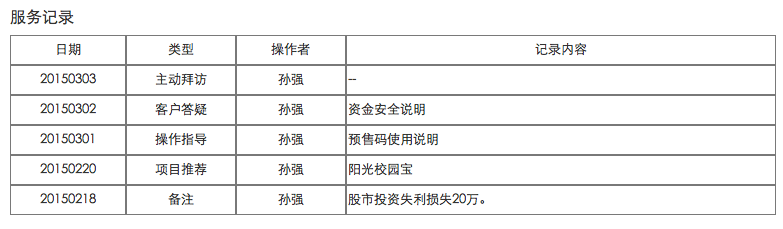
点击取消,回到客户标签页面,不保存结果

【服务记录页面】



图5. 22 服务记录页面

快速记录（同【我的工作台】-【服务速记】，只是客户字段不需填写），服务记录，列表展示服务记录的详情，

图图5. 23 服务记录详情

### 5.2.3 销售管理

基于原【销售管理】，展示销售与项目信息

主要页面与功能点

销售管理页面，“销售项目”改为“在售项目”，其他不变。项目详情页，修改为



图5. 24 销售管理页面

预售创建与记录，整合为预售管理模块

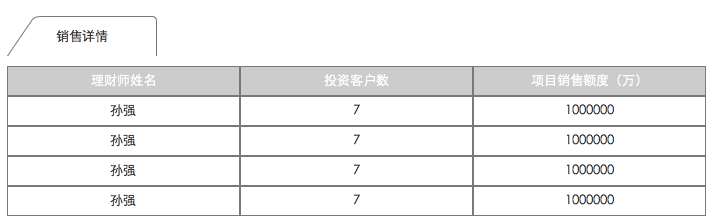
图5. 25 整合后的预售管理模块

新增销售详情模块,展示此项目下该bd的销售记录



图5. 26 销售详情

新增销售排名模块

图5. 27 销售排名

列表展示该项目下，该门店所有BD的销售详情

新增 产品分享



图5. 28 产品分享模块

模块推荐语由BD填写，不可空；点击【微信分享】，生成二维码，扫描二维码，获得可以分享给指定微信好友或分享到朋友圈的图文消息，消息包括项目卡片信息+BD的推荐语；具体实现方案待陆扬调研后确定；点击【微博分享】，新tab跳转到微博转发页面，转发对象为官微上对该项目的介绍微博，转发文案为BD填写的推荐语。

新增销售趋势模块



图5. 29 销售趋势

以气泡图的方式，展示每个tag用户的投资额加总情况，刺激BD多添加标签

### 5.2.4 结算管理



图5. 30 项目状态的调整

修改项目状态为财务打款状态，需配合相应财务后台的修改。

### 5.2.5 理财师课堂

主要用来做理财师的培训基地、做理财师的妙计锦囊做理财师的操作指南

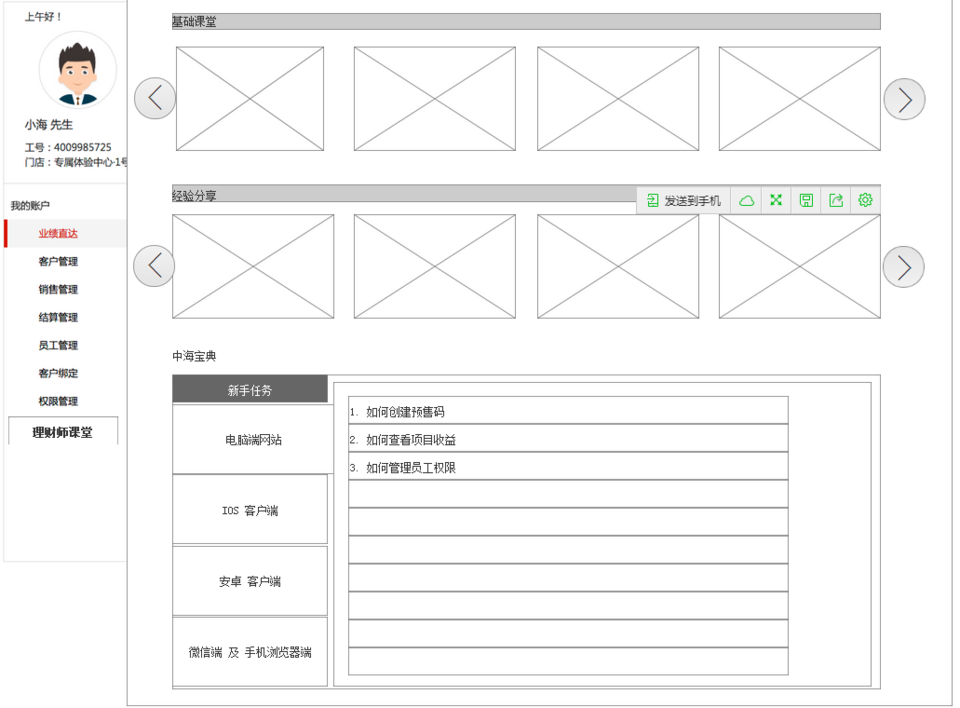


图5. 31 理财师课堂页面

1）基础课堂：每屏展示4个,按切换按钮,每次切换四个；最左端只能往右边切换

2）经验分享：每屏展示4个,按切换按钮,每次切换四个，最左端只能往右边切换

3）中海宝典

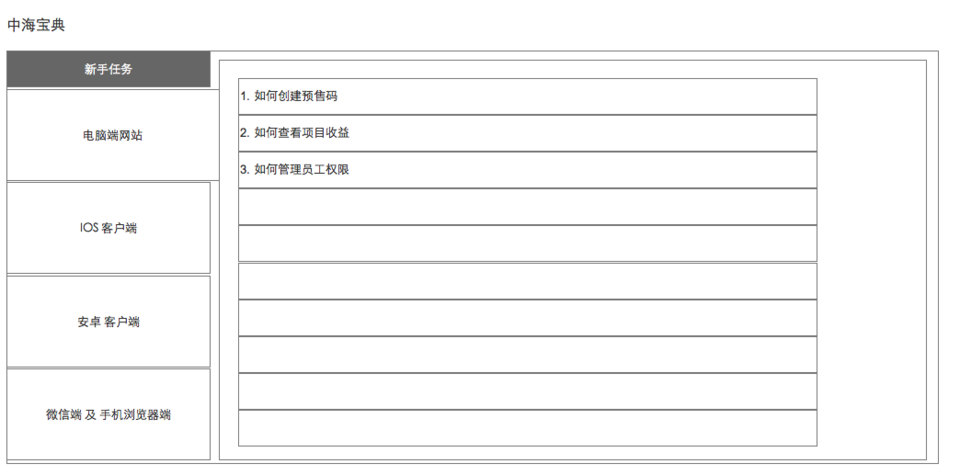


图5. 32 中海宝典

主要用来是说明电脑端网站的各种操作问题合集、IOS客户端的各种操作问题合集、安卓客户端的各种操作问题合集、微信端及手机浏览器端的各种操作问题合集、点击每个都是web文章页，内容待填充。

考虑到渠道本来没有做任务的习惯，去掉做新手任务的想法，直接把新手任务做成Q&A展现

## 5.3 平台管理功能

平台管理员通过audit自用系统进行系统维护，同时用来实现对交易进行中间操作。例如用户进行线下充值需要平台进行审核，审核通过后才可以进行账户余额的变更；提现操作，需要用户发起提现申请，由后台管理人员进行审核，并进行第三方打款操作；平台需要上新的项目时，需要在后台进行上单操作，填写项目的信息，并进行财务、风控和法务三道审核。因而audit中包含了提现审核、门店管理、项目管理、充值审核等功能。



图5. 33 平台管理登录页面



图5. 34 中海平台管理主要功能

## 5.4 本章小结

本章分流程讲了金融网络销售系统的设计方案以及实现手段。分成三个 大的部分来讲，用户应用软件端、理财门店端以及平台管理后台。每一部分都按业务流程分开来表述。其中应用软件包含了7个流程，门店管理包含了5个模块。

# 第六章 系统实现展示与测试

应用软件整体框架是MVC， MVC全称是Model View Controller，也就是模型（Model）、视图（View）、控制器（controller）。用到的设计模式是：单例模式，观察者模式。安卓端用到了Intent意图技术、Json数据处理技术、Handler 消息机制、HttpClient 相关技术使用、SharePreference存储技术。iOS端用到了objective-c开发语言，最新的sdk。Android用到了四大组件：Activity，Service，Content Provider，Broadcast receiver。

## 6.1 系统实现

### 6.1.1 平台管理实现

通过管理后台，平台管理人员可以进行上单管理、素材管理、门店管理以及充值提现审核。上单包含录入订单信息、风控审核、财务审核。进过三道审核的项目，会显示在应用软件和PC主站上。

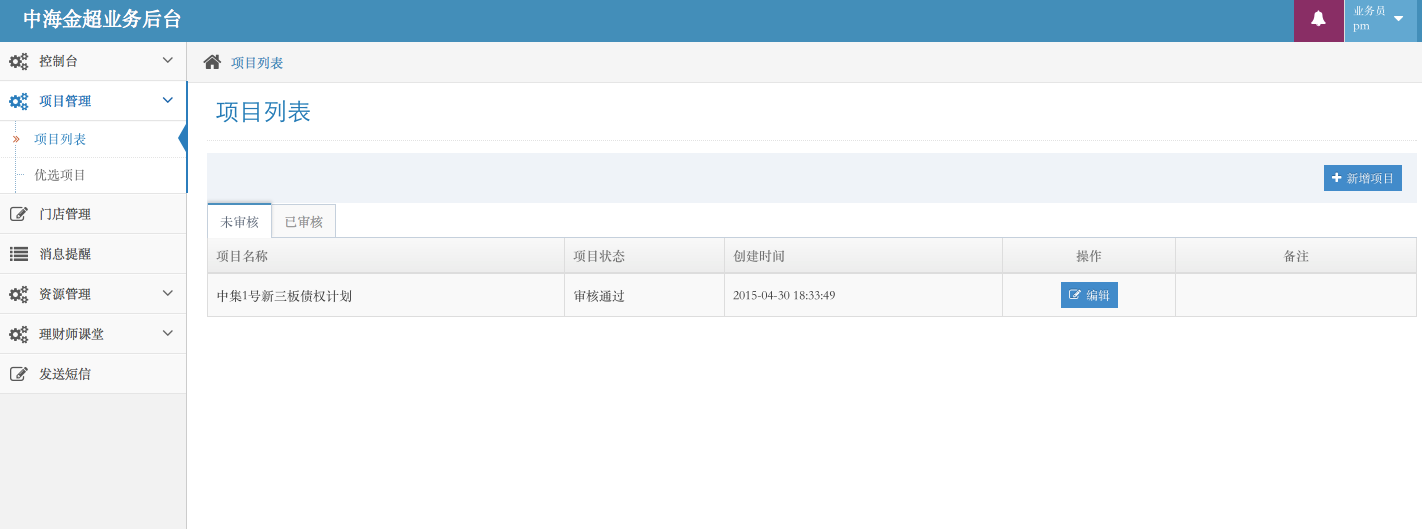


图6. 1 平台实现截图

通过过下图可以项目订单信息。

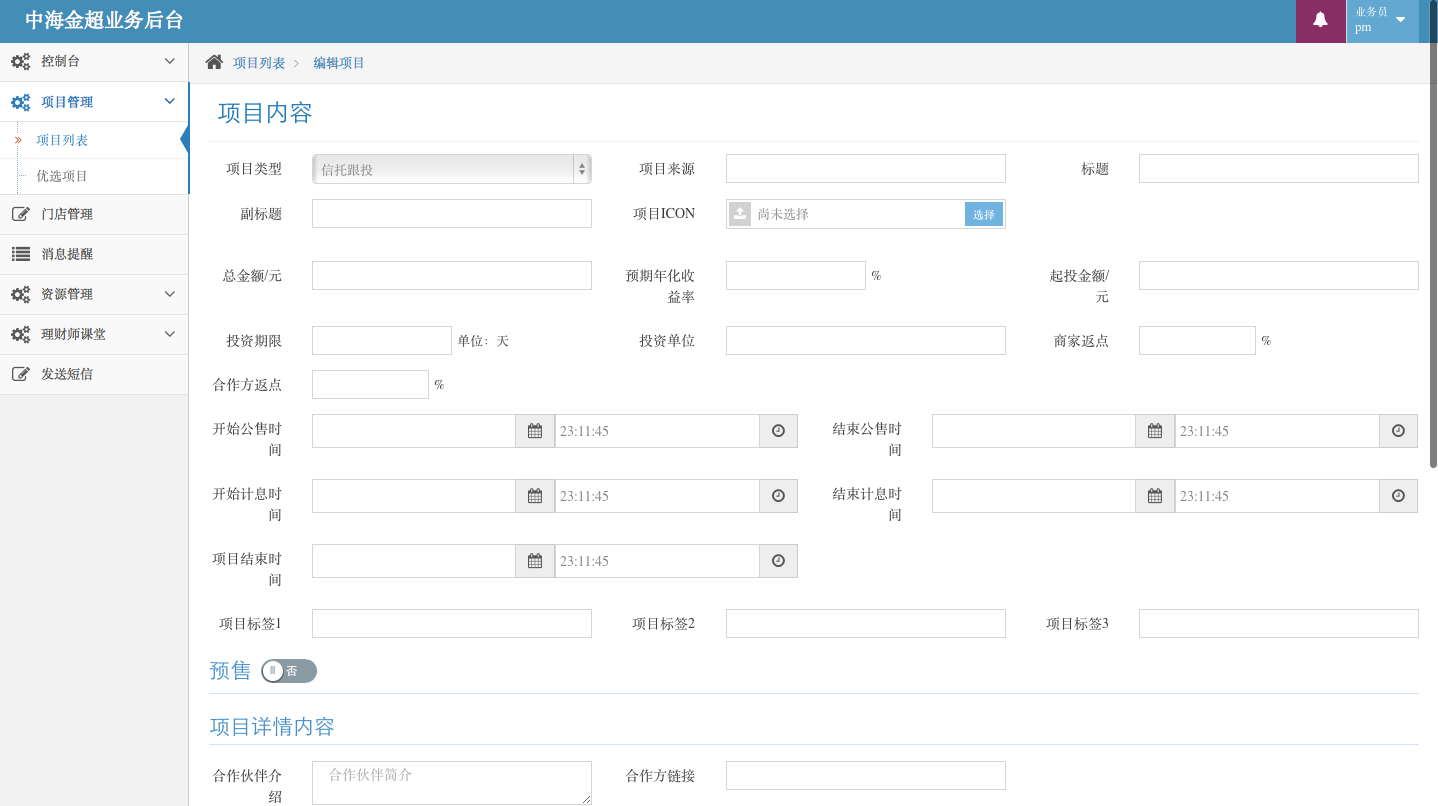


图6. 2 上单管理

通过管理后台可以对应用软件的webview内容进行更新。

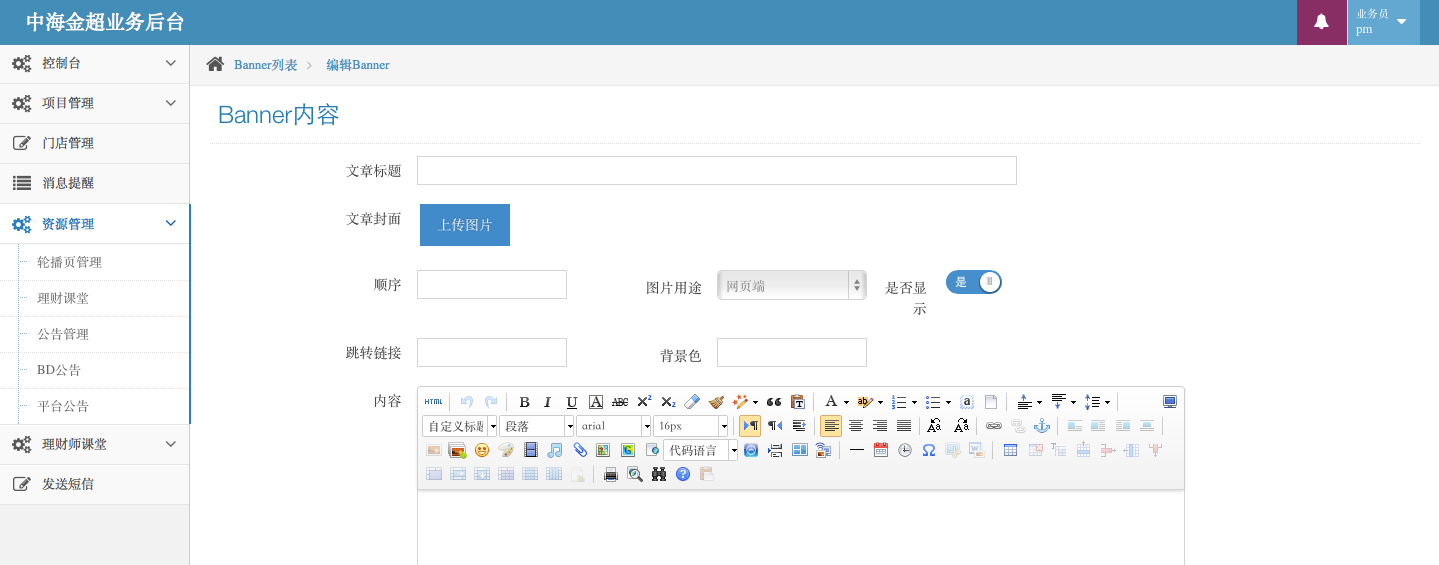


图6. 3 素材管理后台

通过管理后台可以进行提现的管理，进行审核和驳回操作。



图6. 4 提现管理后台

### 6.1.2 用户端实现

应用软件端的所有功能是围绕让用户方便快捷的完成理财操作。客户端是指某互联网金融企业平台的投资客户使用的产品集合体，包括注册与登录流程、项目展示流程、购买与订单管理流程、收益查看流程、支付与提现流程、应用软件设置流程等。C端的移动端产品包括android 应用软件、IOS 应用软件和mobile-web 3个板块。

1）注册与登录流程

打开应用软件是登录页，输入用户名和密码后，校验通过用户可以直接登录。



图6. 5 应用软件登录界面

如果用户未注册，可以点击注册进行注册。注册分为四个步骤。

图6. 6 登录流程截图（a、b、c、d）

2）项目展示流程

项目展示主要的功能是让用户进入应用软件后，由粗到细的看到项目，通过首页的精选项目和项目list页以及项目详情页来完成。

图6. 7 应用软件首页

点击项目进度或查看详情会进入项目详情页

图6. 8 项目详情页

点击项目介绍会进入更加细致的项目详情中，会展示诸如合作方信息、购买合同模板以及还款时间节点的信息。点击投资记录会展示购买信息，为保护客户隐私，用户名都经过加密处理。

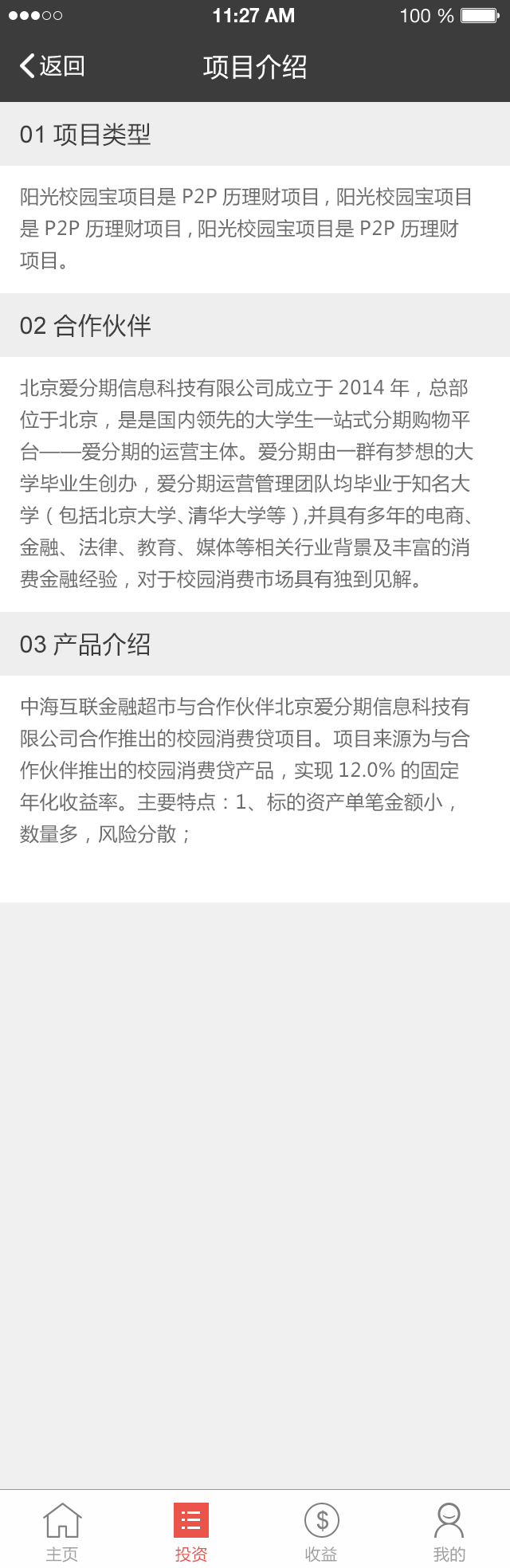
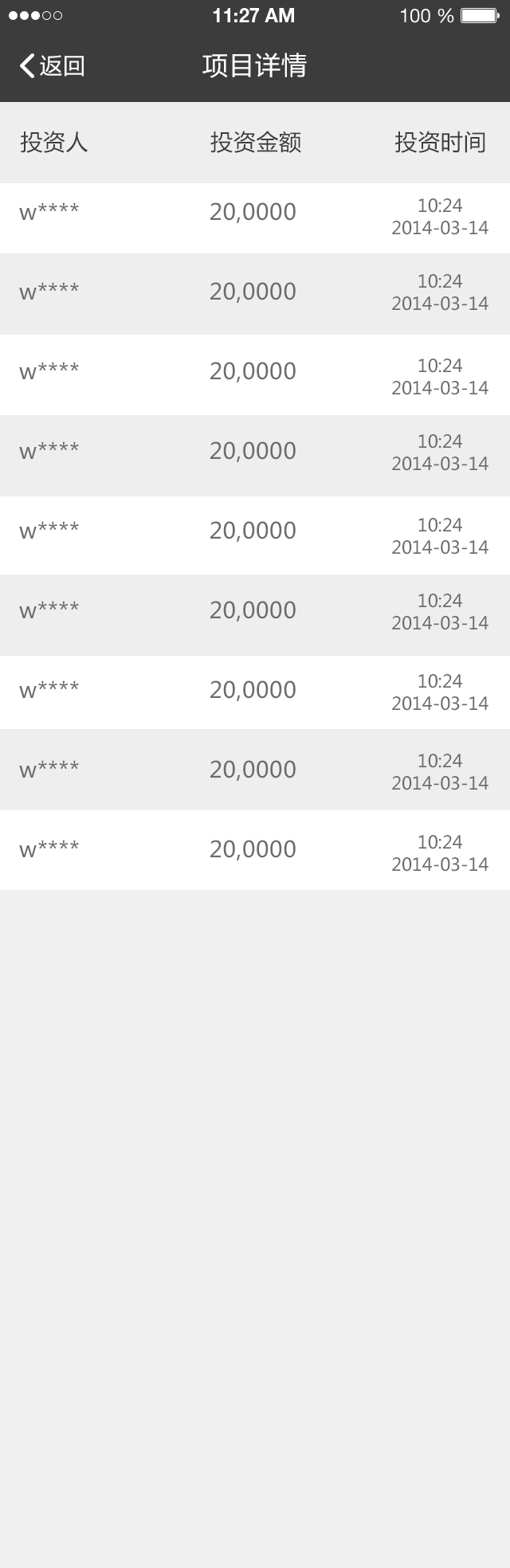
 

图6. 9 项目介绍页及交易记录

3）收益查看流程

这个流程主要以展示用户的资产，管理和订单相关的信息，如未支付订单，绑定的提现银行卡。通过此流程，用户可以把控自己账户的所有资产信息。

图6. 10 收益查看页面

账户总资产，等于在投资金+可用余额+冻结资金，保留两位小数点，单位为万元；可用余额，项目的有效充值和已到账收益中除去提现手续费，冻结资金及在投资近所剩下的金额；在投金额，目前出于已付款和已结项之间的投资累计；冻结资金，提现申请的金额；累计投资，从注册开始到现在的投资金额。

账户设置流程

账户设置包含了实名认证、登录密码、提现密码、密保问题、关于中海、帮助中心、版本信息、检查更新等。



图6. 11 账户设置页面

### 6.1.3 门店管理的实现

理财师是平台和用户接触最多的一个角色，理财师管理后台包含我的工作台、客户管理、销售管理、员工管理、权限管理、理财师学院以及消息模块。



图6. 12 门店管理后台截图



图6. 13 提现管理后台截图

## 6.2 系统测试

### 6.2.1 应用软件测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能点** | | **测试方法** | **预测** | **实际** |
| **登录** | |  |  |  |
| 验证用户名和密码是否为空 | | 测试数据进行手测 | 正常 | 正常 |
| 验证输入用户名或手机号是否存在 | | 测试数据进行手测 | 正常 | 正常 |
| 验证输入密码是否正确 | | 测试数据进行手测 | 正常 | 正常 |
| 验证按钮点击事件（是否成功登录，跳转） | | 测试数据进行手测 | 正常 | 正常 |
| **注册** | | 测试数据进行手测 | 正常 | 正常 |
| 注册第一步（手机号码） | | 测试数据进行手测 | 正常 | 正常 |
| 验证手机号码是否已注册 | | 测试数据进行手测 | 正常 | 正常 |
| 验证手机号码格式是否合法 | | 测试数据进行手测 | 正常 | 正常 |
| 验证“获取验证码”按钮是否可点击 | | 测试数据进行手测 | 正常 | 正常 |
| 验证验证码是否正确 | | 测试数据进行手测 | 正常 | 正常 |
| 验证按钮点击事件（是否成功跳转下一步） | | 测试数据进行手测 | 正常 | 正常 |
| 注册第二步（用户名&密码） | | 测试数据进行手测 | 正常 | 正常 |
| 验证用户名是否已存在 | | 测试数据进行手测 | 正常 | 正常 |
| 验证用户名格式是否合法 | | 测试数据进行手测 | 正常 | 正常 |
| 验证密码格式是否合法 | | 测试数据进行手测 | 正常 | 正常 |
| 验证按钮点击事件（是否成功跳转下一步） | | 测试数据进行手测 | 正常 | 正常 |
| 注册第三步（地址&理财师） | | 测试数据进行手测 | 正常 | 正常 |
| 验证理财师手机号是否存在 | | 测试数据进行手测 | 正常 | 正常 |
| 验证理财师手机号是否为空（可为空） | 测试数据进行手测 | | 正常 | 正常 |
| 注册第四步（实名认证） | | 测试数据进行手测 | 正常 | 正常 |
| 验证跳过按钮是否点击 | | 测试数据进行手测 | 正常 | 正常 |
| 验证姓名和身份证号是否为空 | | 测试数据进行手测 | 正常 | 正常 |
| 验证姓名和身份证号是否正确 | | 测试数据进行手测 | 正常 | 正常 |
| 验证按钮点击事件（是否成功跳转下一步） | | 测试数据进行手测 | 正常 | 正常 |
| 注册完成 | | 测试数据进行手测 | 正常 | 正常 |
| 验证按钮点击事件（是否注册成功，跳转） | | 测试数据进行手测 | 正常 | 正常 |
| **项目展示** | |  |  |  |
| 首页字段是否展示完全 | | 后台传数据 | 正常 | 正常 |
| 展示方法是否正常 | | 后台传数据 | 正常 | 正常 |
| 能否跳转项目详情 | | 后台传数据 | 正常 | 正常 |
| 菜单是否可以顺利展开 | | 后台传数据 | 正常 | 正常 |
| 后端数据是否可以正常显示 | | 后台传数据 | 正常 | 正常 |
| 事实数据是否显示正常 | | 后台传数据 | 正常 | 正常 |
| 横切tab切换正常 | | 测试数据进行手测 | 正常 | 正常 |
| 查看合同正常 | | 测试数据进行手测 | 正常 | 正常 |
| 查看投资记录 | | 测试数据进行手测 | 正常 | 正常 |
| 查看收益 | | 测试数据进行手测 | 正常 | 正常 |
| **我的资产页** | |  |  |  |
| 类型是否齐全 | | 测试数据进行手测 | 正常 | 正常 |
| 账户资产页 | | 测试数据进行手测 | 正常 | 正常 |
| 冻结资金显示是否正常 | | 后台传数据 | 正常 | 正常 |
| 账户余额显示是否正常 | | 后台传数据 | 正常 | 正常 |
| 我的订单 | | 测试数据进行手测 | 正常 | 正常 |
| 交易记录 | | 测试数据进行手测 | 正常 | 正常 |
| 筛选功能是否正常 | | 测试数据进行手测 | 正常 | 正常 |
| 交易记录的类型是否正确 | | 测试数据进行手测 | 正常 | 正常 |
| 交易类型是否哦按逆序排列 | | 测试数据进行手测 | 正常 | 正常 |
| 理财师页面 | | 测试数据进行手测 | 正常 | 正常 |
| 理财师信息是否展示正常 | | 测试数据进行手测 | 正常 | 正常 |
| 拨打电话功能是否正常 | | 测试数据进行手测 | 正常 | 正常 |
| 提现银行卡 | | 测试数据进行手测 | 正常 | 正常 |
| 银行卡信息是否展示正常 | | 测试数据进行手测 | 正常 | 正常 |
| 充值页面 | | 测试数据进行手测 | 正常 | 正常 |
| 填入金额是正常 | | 测试数据进行手测 | 正常 | 正常 |
| 提交是否正常 | | 测试数据进行手测 | 正常 | 正常 |
| 提现页面 | | 测试数据进行手测 | 正常 | 正常 |
| **账户设置** | |  |  |  |
| 验证按钮点击事件（是否修改  成功，跳转） | | 测试数据进行手测 | 正常 | 正常 |
| 实名认证 | | 测试数据进行手测 | 正常 | 正常 |
| 登录密码 | | 测试数据进行手测 | 正常 | 正常 |
| 修改登录密码 | | 测试数据进行手测 | 正常 | 正常 |
| 找回密码 | | 测试数据进行手测 | 正常 | 正常 |
| **更多** | |  |  |  |
| 能否查看版本信息 | | 测试数据进行手测 | 正常 | 正常 |
| 手势密码 | | 测试数据进行手测 | 正常 | 正常 |
| 修改手势密码 | | 测试数据进行手测 | 正常 | 正常 |
| 设置手势密码 | | 测试数据进行手测 | 正常 | 正常 |

### 6.2.2 后台功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| **功能点** | **测试方法** | **预测** | **实际** |
| **注册** |  |  |  |
| 门店手机号码校验 | 手动测试 | 正常 | 正常 |
| 登录密码设置 | 手动测试 | 正常 | 正常 |
| 地址填写校验 | 手动测试 | 正常 | 正常 |
| 姓名重复校验 | 手动测试 | 正常 | 正常 |
| **登录** |  |  |  |
| 用户名校验 | 手动测试 | 正常 | 正常 |
| 验证码校验 | 手动测试 | 正常 | 正常 |
| 登录密码校验 | 手动测试 | 正常 | 正常 |
| 找回密码入口 | 手动测试 | 正常 | 正常 |
| 我的工作台 | 手动测试 | 正常 | 正常 |
| 业绩速览数据显示正常 | 手动测试 | 正常 | 正常 |
| 在线项目更多入口正常 | 手动测试 | 正常 | 正常 |
| 点击消息进入消息列表 | 手动测试 | 正常 | 正常 |
| 点击更多跳转服务记录 | 手动测试 | 正常 | 正常 |
| 添加服务类型是否正常 | 后台传输数据 | 正常 | 正常 |
| **客户管理** |  |  |  |
| 展示用户列表数据是否全 | 后台传输数据 | 正常 | 正常 |
| 点击管理是否跳转用户详情 | 后台传输数据 | 正常 | 正常 |
| 点击新增用户跳转添加列表 | 后台传输数据 | 正常 | 正常 |
| 搜索框姓名sug提示 | 后台传输数据 | 正常 | 正常 |
| 客户详情页面添加标签 | 后台传输数据 | 正常 | 正常 |
| 通过标签筛选用户 | 后台传输数据 | 正常 | 正常 |
| 风险问卷入口是否正常 | 后台传输数据 | 正常 | 正常 |
| 实名认证接口是否正常 | 后台传输数据 | 正常 | 正常 |
| 投资历史展示正常 | 后台传输数据 | 正常 | 正常 |
| 理财师添加tag | 后台传输数据 | 正常 | 正常 |
| 风险问卷生成tag | 后台传输数据 | 正常 | 正常 |
| **销售管理** |  |  |  |
| 展示销售项目 | 后台传输数据 | 正常 | 正常 |
| 点击管理跳转销售详情 | 后台传输数据 | 正常 | 正常 |
| 预售和非预售区别展示预售模块 | 后台传输数据 | 正常 | 正常 |
| 展示销售记录 | 后台传输数据 | 正常 | 正常 |
| 展示销售排名 | 后台传输数据 | 正常 | 正常 |
| 展示报单功能 | 后台传输数据 | 正常 | 正常 |
| 报单中姓名sug功能 | 后台传输数据 | 正常 | 正常 |
| 报单手机号链级填入的功能 | 后台传输数据 | 正常 | 正常 |
| 报单录入银行信息的功能 | 后台传输数据 | 正常 | 正常 |
| 取消错误报单操作 | 后台传输数据 | 正常 | 正常 |
| **结算管理** |  |  |  |
| 展示项目列表 | 后台传输数据 | 正常 | 正常 |
| 展示项目信息 | 后台传输数据 | 正常 | 正常 |
| 筛选tab切换正常 | 后台传输数据 | 正常 | 正常 |
| 本月销售数据展示正常 | 后台传输数据 | 正常 | 正常 |
| **员工管理** |  |  |  |
| 展示员工列表 | 后台传输数据 | 正常 | 正常 |
| 展示员工信息 | 后台传输数据 | 正常 | 正常 |
| 点击编辑进入编辑页面 | 后台传输数据 | 正常 | 正常 |
| 点击删除去除员工信息 | 后台传输数据 | 正常 | 正常 |
| **权限管理** |  |  |  |
| 展示权限 | 后台传输数据 | 正常 | 正常 |
| 勾选权限功能 | 手动测试 | 正常 | 正常 |
| 点击保存更新数据库 | 手动测试 | 正常 | 正常 |
| **账户设置** |  |  |  |
| 更改当前密码 | 手动测试 | 正常 | 正常 |
| 上传个人头像 | 手动测试 | 正常 | 正常 |
| 更新住址 | 手动测试 | 正常 | 正常 |

## 6.3 本章小结

本章对最终实现的成果进行展示，包含用户移动端应用软件以及门店管理后台的主要的业务流程主要界面，并对对系统进行了测试，测试包含了易用性测试、压力测试兼容性测试，目前所有已实现的功能都已正常运行。

# 第七章 总结与展望

## 7.1 课题总结

互联网金融的在占领了诸多报纸头条的时候，线上和线下的竞争也给广大用户带来了切实的好处。本课题基于实际项目：某互联网公司，欲在看似红海的互联网理财中，寻找差异化，找到未被满足的用户需求。目前二三线城市的中年人被互联网金融边缘化，但实际上他们是高质量的客户，本系统设计就是为了满足他们的理财需求。

本系统围绕让用户在理财师的帮助下完成理财产品的购买，因而涉及到三个相关方：用户、理财师和平台管理人员，三个角色分别对应着移动端应用软件、理财师管理后台和平台管理后台。经过了需求分析、系统架构分析、功能模块分析、页面设计和需求实现阶段，将课题一步步分解，逐渐深入，其中运用到了易用性、通用性、可拓展性的原则。应用软件完成后进行了多轮测试，最终发布上线。系统与预期相符，实现了如下功能：

应用软件端包含：项目展示功能、下单和支付功能、收益查看功能、充值提现功能、账户设置功能。

理财师管理后台包含：销售管理、业绩管理、客户管理、结算管理、员工管理、权限管理。

平台管理后台包含：建立门店、项目管理、充值管理、提现申请管理、报单审核。

本文通过分析实际业务情况，提出解决方案，并最终运用软件开发的流程实现了设计方案。将需求分析、系统架构和软件开发应用在了互联网公司的实际业务中，为互联网公司及传统金融机构的互联网化提供了思路。

## 7.2 不足和展望

某互联网金融企业的应用软件在加班加点中，最终与用户碰面，虽然我们力求完美，但限于时间和能力，还是有很多问题。在后期的迭代中，这些将成为我们首要的优化点。

第一、需求实现过称中，很多方案过于复杂，浪费了开发资源，这些是产品设计中应该避免的事。

第二、需求设计中需求的优先级没有把握好，软件开发应该短周期迅速迭代的过程。一期的功能过于庞杂。

第三、交易密码只形成了文档，还没有落于实践。

在移动互联网的浪潮中，我们是一个刚起帆的弄潮儿。当各种互联网+大行其道的时候，机会与挑战都在扑面而来，移动应用软件的上线只是一个开始，更多的工作还要去展开。

第一、数据库的设计是否合理，当大量用户访问时，是否会出现读取速度过慢。

第二、引入自动化测试，在极端的情况下是否会出现crash。

第三、对于新技术的尝试，可以提高开发效率，例如用phonegap可以快速封装应用软件，比传统的native开发方法周期缩短三分之一。

# 参考文献

1. [王红英](http://www.cnki.net/kcms/detail/search.aspx?dbcode=CCND&sfield=au&skey=%e7%8e%8b%e7%ba%a2%e8%8b%b1&code=" \t "_blank).民间银行破壳而出 互联网银行备受关注.企业家日报,2014,008:1-3

1. [洪偌馨](http://www.cnki.net/kcms/detail/search.aspx?dbcode=CCND&sfield=au&skey=%e6%b4%aa%e5%81%8c%e9%a6%a8&code=" \t "_blank).多部门再探互联网金融 P2P领衔“灰名单”.第一财经日报,2013,A01:4-9

1. [杨晋亭](http://www.cnki.net/kcms/detail/search.aspx?dbcode=CCND&sfield=au&skey=%e6%9d%a8%e6%99%8b%e4%ba%ad&code=31742271;" \t "_blank).浅析互联网金融. 科学导报,2014,C01: 2-9
2. CNNIC:2014年第35次中国互联网络发展状况报告.http://www.199it.com/archives/257541.html
3. 葛春晖.多头高位止盈 期债减仓重挫.中国证券报: 衍生品/对冲,A15:1-10
4. 财经网.余额宝用户数突破8100万规模或已突破5000亿元[EB/OL].[http://money.163.com](http://money.163.com/)
5. 郑联盛.中国互联网金融:模式、影响、本质与风险.国际经济评论,2014,05:1-3
6. 申变云.TC公司P2P网贷平台可行性分析及设计研究[硕士学位论文]. 四川:西南交通大学, 2014
7. 龚明华.互联网金融:特点、影响与风险防范.新金融,2014,02:56-59
8. 叶湘榕.借贷的模式风险与监管研究. 金融监管研究, 2014,3:71-83
9. 谢平,邹传伟.互联网金融模式研究.金融研究,2012,12:11~22
10. 红象金融. 2015年互联网金融理财用户调研报告. http://www.doc88.com/p-0562611411063.html
11. 徐丰.基于.NET的跨馆图书管理系统的设计与实现 [硕士学位论文]. 北京:北京邮电大学,2012
12. 宋海曦.基于Android平台的用户分级拨号软件开发 [硕士学位论文]. 北京:北京邮电大学,2013
13. 陈平.基于是Spring的轻量级Web框架的研究与设计[D] .江苏镇江:江苏大学,2005
14. 刘新平.基于移动GIS的土地动态巡查监管系统研究与应用[硕士学位论文]. 南京:南京师范大学,2011
15. 朱树原.工作流技术在纪检监察信息管理系统中的应用[硕士学位论文].天津:天津大学,2012
16. 李晓禹.基于SOA的设备管理信息系统平台的研究与实现[硕士学位论文].南京:南京大学,2011
17. 方国.基于J2EE平台的软件设计模式研究与应用[硕士学位论文].北京:华北电力大学（北京）,2004
18. Omori T , Mochizuki T , et al. Emergence of Sysbolic Behavior f romLike M emory w ith Dynamic Att ent ion[ J] . Neural Networks, 1999,( 12) : 1157-1172
19. Hui Zhang, Yu Song, Han-tao Song. [User Modeling 2007](http://d.scholar.cnki.net/Detail/SSBD_U/SSBD00000536874)[B]. Lecture Notes in Computer Science, 2007.
20. André C. Nácul, Marcello Lajolo, Tony Givargis. [Applications of Specification and Design Languages for SoCs](http://d.scholar.cnki.net/Detail/SSBD_U/SSBD00000533865)[B]. Springer Netherlands, 2006.
21. Murdock Jr B B.T ODAM2: A Model for the St orage an d Ret rieval ofItem, Associative and S erial order Informat ion[ J] . Psychological Review , 1993, 100( 2) :183-203.
22. 周宇.基于J2ME与J2EE平台的移动商务应用及安全性研究[硕士学位论文].贵州:贵州大学,2006
23. 姜多.政府采购管理系统设计[硕士学位论文].长春:吉林大学,2011
24. 李军强.源代码评估系统的研究与开发[硕士学位论文].北京:北京化工大学,2011
25. 史燕.中学教务管理信息系统的分析与设计[硕士学位论文].厦门:厦门大学,2011
26. 吴依蔓.西部畜牧产品电子交易平台的设计与实现[硕士学位论文].重庆:重庆大学,2011
27. 谯兴国.襄樊市劳动系统工资管理软件的设计与实现[硕士学位论文].成都:电子科技大,2013
28. 卢展晴.基于MVC的新闻考核管理系统的设计与实现[硕士学位论文].广州:华南理工大学,2011
29. 刘越洋.大学生活动信息交流平台的设计与实现[硕士学位论文].厦门:厦门大学,2013
30. 淡欣.基于J2EE平台MVC模式的健康评估与预测管理系统[硕士学位论文].成都:电子科技大学,2011
31. 张琳.基于Lucene的电子公文检索系统的研究与实现[硕士学位论文].沈阳:沈阳理工大学,2013
32. 王坤.基于J2EE平台Spring MVC框架开发的MIS系统设计与实现[硕士学位论文].上海:华东师范大学,2008
33. 张玲.基于EJB及SPRING技术的B2B电子商务平台的设计与实现[硕士学位论文].沈阳:东北大学,2008陈静,王莉娜. 基于SSH2的一站式政府门户网站的应用研究. 江苏教育学院学报:自然科学版, 2002,38(4):47-50
34. 廖华雄.SSH技术在设备缺陷管理系统中的应用研究 [硕士学位论文].长沙:湖南大学,2012
35. 刘晓清.图书管理系统设计与开发[硕士学位论文].成都 电子科技大学,2012
36. 杨艳梅,朱养鹏. 长庆油田物料管理库存优化系统的分析与设计.科技世界,36:24-28
37. 曾雷.轨交优先公交最佳路径建模及其应用研究[硕士学位论文].上海:东华大学,2010
38. 张志鸿. 基于ITIL的商业银行IT运维体系构建[硕士学位论文].长沙:湖南大学,2011
39. 徐晶卉.网络借贷面临大浪淘沙. 文汇报,2013,009:1-3
40. CNNIC：我国手机网民规模至85.8% 手机应用爆发<http://it.people.com.cn/n/2015/0723/c1009>

-27350708.html

1. 詹涛,周兴社,符宁,杨刚.软件能力可信研究综述.小型微型计算机系统,2008,05:5-9
2. 陈坤.嵌入式数字视频录像控制系统的设计与实现[硕士学位论文].湖北:华中科技大学,2005
3. 李海峰.一种基于云计算的软件服务应用平台设计与实现[硕士学位论文]广州:中山大学,2013

# 致谢

在本文即将结束之际，研究生生活也接近尾声,回头看,感慨万千。一路走来，要感谢的人很多。

首先要对我的导师李杰教授表示真诚的感谢！在我读研期间，李老师不仅传授了我知识，同时指引我们正确的成长道路，在那关键的节点，老师总会给我们真诚的建议。李老师严谨治学及平和的待人态度给了我很大的影响。在本次论文期间，李老师也不断的询问我进度，以及需要解惑的问题，没有李老师耐心的督导，我的论文不可能顺利的完成。

感谢深圳黑方石金融服务有限公司，在我实习的近1年的时间里，在这里收获了很多，从待人接物到知识实践，都让我提升了一个高度。感谢我的同事们，与他们的交流，让我取长补短，成就了更好的自己。

感谢北京大学软件与微电子学院，在这里学习了3年，在我身上有了深深软微印记，我因成为一个软微人而感到庆幸，同时我也会努力工作让北大软微因为培养了我们而觉得自豪。

感谢我们的同学们，很多个苦恼和烂漫的日子，我们都一起走过，感谢偶遇你们相伴。

最后，衷心地感谢参与本文评审的各位专家教授学者，感谢您们能在百忙之中抽出宝贵时间审阅本文。

北京大学学位论文原创性声明和使用授权说明

**原创性声明**

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

论文作者签名： 日期： 年 月 日

**学位论文使用授权说明**

(必须装订在提交学校图书馆的印刷本)

本人完全了解北京大学关于收集、保存、使用学位论文的规定，即：

* 按照学校要求提交学位论文的印刷本和电子版本；
* 学校有权保存学位论文的印刷本和电子版，并提供目录检索与阅览服务，在校园网上提供服务；
* 学校可以采用影印、缩印、数字化或其它复制手段保存论文；
* 因某种特殊原因需要延迟发布学位论文电子版，授权学校□一年/□

年/□三年以后，在校园网上全文发布。

(保密论文在解密后遵守此规定)

论文作者签名： 导师签名：

日期： 年 月 日