

**本科毕业设计(论文)**

题 目**：** 资产管理公司业务管理系统

专题题目：

学 院： 软件学院

专 业： 软件工程 班 级： 13网通

学 号： 13222115 学 生： 王健

指导教师： 柴政 职称： 讲师

指导教师： 职称：

时间： 2017年5月10日

**江 西 理 工 大 学**

**本 科 毕 业 设 计（论文）任 务 书**

**软件学院 软件工程 专业 2013级（2017届）网络通信1班 13222115 王健**

**题 目：** 资产管理公司员工管理与业务系统

**专题题目**

**原始依据**

**工作基础**

与启辰资产管理公司的总经理接洽，对方提出了一系列期望，期望用的方便，同时具有交互性。

* **研究条件**

该套系统将在Mac OS Sierra上开发，开发工具为eclipse，JDK1.7

* **应用环境**

该套系统最终将部署到阿里云ECS中。云服务器采用Linux系统，一核2G内存，Tomcat作为容器。

* **工作目的**

**主要内容和要求：**项目突出业务与管理，公司的业务也将在应用上完成，并且录入数据库，通过分析试图，了解业务的变化，通过对客户实现管理。

* **研究内容**

主要解决公司的三大问题，信息的管理，列如员工的信息，客户的信息；业务的实现，跟进业务的流程；数据的展示，数据视图化。

* **主要技术指标与技术参数**

Bootstrap提供优美的界面，MongoDB作为稳定的数据库，Spring-data 提供可靠的数据库事务。

* **具体要求**

独立完成本项目系统研制过程，系统能够顺利运行并且部署，验收，并独立完成论文报告。

**日程安排：**

2017/2/10-2017/2/21 项目需求分析

2017/2/21-2017/2/28 UML设计

2017/3/10-2017/3/25 视图设计

2017/3/26-2017/4/25 业务逻辑的实现

2017/4/26-2017/5/5 项目的测试与调试

2017/5/6-2017/5/10 项目的部署与应用

2017/5/11-2017/5/20 完成毕业设计论文的定稿

**主要参考文献和书目：**

[1]Bruce Eckel.《Thinking in Java》．北京：机械工业出版社，2007

[2] Gary Cornell《JAVA核心技术》.北京:电子工业出版社, 2011

[3] 霍多罗夫.《MongoDB权威指南》.北京：人民邮电出版社,2011

[4]徐涛.《深入理解Bootstrap》北京：.机械工业出版社,2014

[5]《jQuery教程》. W3School

[6]Nicholas C.Zakas. 《JavaScript高级程序设计》.北京.人民邮电出版社

[7]巴萨姆.《Head First Servlet JSP》.中国电力出版社.2006

[8]Paul Deck.《SpringMVC学习指南》.北京.人民邮电出版社.2015

[9]百度. 《ECharts》.百度公司

[10]程晓明.《深入理解Java内存模型》

**指导教师**（签字）**：**

**年 月 日**

注：本表可自主延伸，各专业根据需要调整。

**江 西 理 工 大 学**

**本 科 毕 业 设 计（论文）开 题 报 告**

**软件学院 软件工程 专业 2013级（2017届）网络通信1班 13222115 王健**

**题 目：**资产管理公司员工管理与业务系统

**专题题目：**

**本课题来源及研究现状：**

* **课题来源：**

随着互联网的高速发展，各个企业都开始追求信息化，自动化。通过计算机科技，解放生产力，使员工更加专注的投入到自身的业务中去，从而使得公司在行业竞争中处于优势地位，因此一个优秀的业务管理系统是必要的，本人应上海起辰资产管理有限公司的要求，于2017年2月为其设计并实现其公司的公司员工管理系统与对外业务系统，两者系统合二为一，合称为“起辰资本业务系统”。

* **研究现状：**

现已完成需求分析，项目架构（基于SpringMVC），数据库设计（MongoDB），UML设计，接下来会完成视图设计（Html&JSP），前后端交互（Servlet），然后进行项目测试，调试Bug，优化视图加载速度，最终将项目打成war包部署进“阿里云”ECS服务器。

**课题研究目标、内容、方法和手段：**

* **研究目标：**

对于该公司来，其最关心的是实现定时提醒客户交付利息，员工的工资管理，以及员工其手中的资产管理，其迫切希望实现该公司目前所需求的各种功能，帮助其摆脱传统的Excel表格时代，人工查询时代，走向自动化，信息化时代，并且在软件设计初期为之后的更新功能留有余地，实现高耦合低内聚，完全按照软件开发流程进行开发系统。

* **研究内容：**

“起辰资本业务系统”分为两大部分。一部分为内部员工管理系统，包含了公司内部职员的个人信息，个人工资（底薪加提成），个人所在部门等；

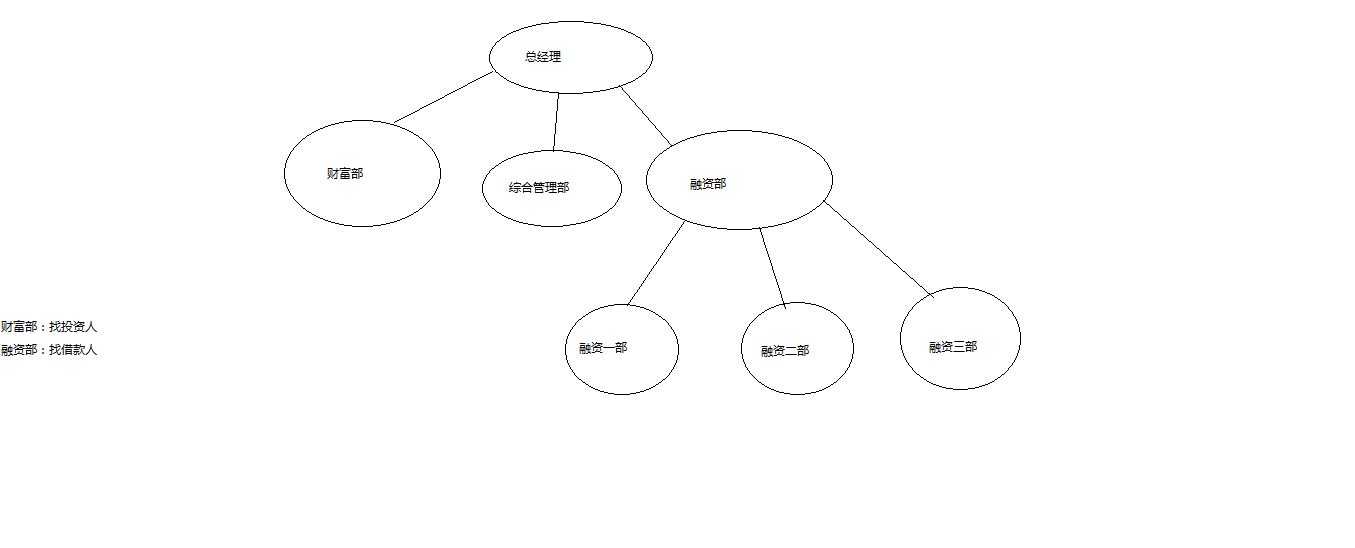


图1.1部门关系图

第二部分为业务系统，其包含了出借信息，每日更新提醒等。

* **研究方法：**

该业务系统采用了Java为主要语言，采用MVC模式进行开发，框架选用Spring+SpringMVC+MongoDB，前端页面Html5+css+javascript+Bootstrap，以eclipse.neno 1为开发工具，采用tomcat8.5为服务器，JDK为1.7版本，服务器搭载在“阿里云”上。

**设计（论文）提纲及进度安排：**

* **设计（论文）提纲：**

第一章 绪论

第二章 相关技术与方法

第三章 系统分析

第四章 系统设计

第五章 系统实现

第六章 系统测试

第七章 总结

* **进度安排：**
* 2017/2/10-2017/2/21 项目需求分析
* 2017/2/21-2017/2/28 UML设计
* 2017/3/10-2017/3/25 视图设计
* 2017/3/26-2017/4/25 业务逻辑的实现
* 2017/4/26-2017/5/5 项目的测试与调试
* 2017/5/6-2017/5/10 项目的部署与应用
* 2017/5/11-2017/5/20 完成毕业设计论文的定稿

**主要参考文献和书目：**

[1]Bruce Eckel.《Thinking in Java》．北京：机械工业出版社，2007

[2] Gary Cornell.《JAVA核心技术》.北京:电子工业出版社, 2011

[3] 霍多罗夫.《MongoDB权威指南》.北京：人民邮电出版社,2011

[4]徐涛.《深入理解Bootstrap》北京：.机械工业出版社,2014

[5]《jQuery教程》. W3School

[6]Nicholas C.Zakas. 《JavaScript高级程序设计》.北京.人民邮电出版社

[7]巴萨姆.《Head First Servlet JSP》.中国电力出版社.2006

[8]Paul Deck.《SpringMVC学习指南》.北京.人民邮电出版社.2015

[9]百度. 《ECharts》.百度公司

[10]程晓明.《深入理解Java内存模型》

**指导教师审核意见：**

**指导教师（**签字）**： 年 月 日**

注：本表可自主延伸

摘 要

随着互联网与IT技术的迅猛发展，人类的日常生活与工作都离不开软件的支持，基于网络的web应用软件成为当前和未来的软件系统开发的主流。而由于pc机的性能快速提高，传统的低交互性程序已无法满足社会的需求，于是“富客户端技术”应运而生，使用该技术的程序被称为“富因特网应用程序”（Rich Internet Applications,RIA），它利用具有强交互性的富客户端技术来为用户提供一个更优质和更全方位的网络体验。

为了提供web应用程序具有更强的交互性，将采用大量的web前端技术，实际上，一个优秀web应用程序也需要依赖于一个优秀的web应用服务器，本文在大量的RIA开发经验基础上，设计并开发某资产管理公司的员工管理与业务系统，采用B／S结构（Browser/Server），并在数据可视化（Data visualization）方面作出了一些努力。由于BootStrap技术在前端页面能显现更加绚丽的效果，以及更好的安全性，所以放弃Html4.01，而使用Html5+css3 技术。一个优秀的RIA是离不开脚本语言的支持，通过ajax技术与后台进行异步交互，javascript语言是目前RIA开发的标准之一。而数据库采用MongoDB，一个介于传统关系数据库与非关系数据库之间的数据库产品，类似于CERN，纽约时报，都采用该数据库。而后台方面， 采用Spring+SpringMVC+MongoDB模式。最终将RIA部署在Apache tomcat web应用服务器上。

关键词：富因特网应用程序（RIA）；B／S结构（Browser/Server）；数据可视化；MongoDB；BootStrap；Spring

**ABSTRACT**

With the rapid development of Internet and IT technology, human daily life and work are inseparable from the software support, web-based web application software to become the mainstream of the current and future software system development. And the rapid performance of the pc machine, the traditional low-interaction program has been unable to meet the needs of society, so "rich client technology" came into being, and the use of the technology application is called "rich Internet applications" Rich Internet Applications, RIA), which uses a highly interactive, rich client technology to provide users with a higher and more comprehensive network experience.

In order to provide web applications with more interactive, will use a lot of web front-end technology, in fact, an excellent web application also need to rely on a good web application server, based on a large number of RIA development experience, And the development of an asset management company's staff management and business systems, the use of B / S structure (Browser / Server), and data visualization (Data visualization) made some efforts. As BootStrap technology in the front page can show more brilliant results, and better security, so give up Html4.01, and the use of Html5 + css3 technology. A good RIA is inseparable from the scripting language support, through ajax technology and asynchronous interaction with the background, javascript language is the current RIA development standards. The database uses MongoDB, a database based on distributed file storage, similar to CERN, the New York Times, are using the database. The background, the use of commonly used Spring + SpringMVC +MongoDB. Finally, the RIA is deployed on the Apache tomcat web application server.

**Key words:** Rich Internet Applications (RIA); B / S structure(Browser/Server); Data visualization; MongoDB;BootStrap;Spring

**目 录**

[第一章 绪论 2](#_Toc482966129)

[1.1 研究背景和意义 2](#_Toc482966130)

[1.2 研究目标 3](#_Toc482966131)

[1.3 论文结构 3](#_Toc482966132)

[1.4 本章小结 3](#_Toc482966133)

[第二章 相关技术与方法 4](#_Toc482966134)

[2.1 架构概述 4](#_Toc482966135)

[2.2 关键技术简介 5](#_Toc482966136)

[2.3 开发工具 6](#_Toc482966137)

[2.4 本章小结 6](#_Toc482966138)

[第三章 系统分析 7](#_Toc482966139)

[3.1 可行性分析 7](#_Toc482966140)

[3.2 需求分析 8](#_Toc482966141)

[3.3 本章小结 12](#_Toc482966142)

[第四章 系统设计 13](#_Toc482966143)

[4.1 系统类分析 13](#_Toc482966144)

[系统基类图： 13](#_Toc482966145)

[4.2 关键业务设计 15](#_Toc482966147)

[4.3 数据库设计 17](#_Toc482966148)

[4.4 本章小结 20](#_Toc482966149)

[第五章 系统实现 21](#_Toc482966150)

[5.1 模块实现 21](#_Toc482966151)

[5.2 本章小结 27](#_Toc482966152)

[第六章 系统测试 28](#_Toc482966153)

[6.1 系统测试综述 28](#_Toc482966154)

[6.2 测试用例 28](#_Toc482966155)

[6.3 测试分析 29](#_Toc482966156)

[6.4 本章小结 29](#_Toc482966157)

[第七章 总结 30](#_Toc482966158)

[附页 小论文 31](#_Toc482966159)

[文章 I. 参考文献 34](#_Toc482966160)

[文章 II. 致 谢 35](#_Toc482966161)

1. 绪论
   1. 研究背景和意义

21世纪初，随着计算机快速普及与网络的不断完善，使用计算机成了人们生活和工作不可或缺的一部分，计算机成了提高公司生产力的最佳方式。通过各种软件的支持，使得公司的管理更加有条理，更加便利。事实证明，科技是提高生产力的有力工具。

“上海启辰资产管理公司”是一家资产借贷公司，提供给客户的优秀的资产管理，客户通过自己的资产公证，抵押，来获得相应额度的资金。由于其客户群愈发增多，传统的excel表格检索已无法满足其的需求，excel表格也无法提供每日提示催息的功能，无法直观的显示其业绩。于是应“上海启辰资产管理公司”的要求，为其公司开发一套服务于企业员工的管理与业务系统，以便帮助该公司提高业务能力，将公司管理简单化。翻看中外的开发书籍，采用Java语言开发无疑是最好的选择。Java语言是目前最流行的高级编程语言，其超过了著名的C语言，C++，一举成为了第三代编程语言的No.1。Java语言用其的简单性，健壮性，可移植性等优秀特性，征服了大量的编程人员，其“一处编译，随处运行”的跨平台的优点，解决计算机平台的不兼容问题，编程人员不在需要为了不同平台进行不同的开发。由于Java语言的开发公司Sun公司秉持着开源的态度，这使得Java这门语言具有强大API①库，为了程序员的开发提供的了很大的便利。目前，Java分为三个版本，JavaME②为支持移动端设计，JavaSE③标准版，JavaEE④企业版。JavaEE其对互联网具有很好的支持，因此将作为开发的版本。当下企业对于应用程序，都要求有很强的交互性，这对于前台有了更高的要求，除了常规的html语言之外，还需大量的脚本语言来帮助进行交互，在后台方面，将采用“MVC”模式，进行开发。通过开发该套系统，已帮助提高公司的业务能力，用优秀的数据视图化展示给用户，使其更加快速抓住数据重点。

1. API（应用程序编程接口，Application Programming Interface），提供开发人员于应用程序基于某软件或硬件得以访问的能力，无需访问源代码，或理解内部工作机制的细节。
2. JavaME（Java Platform，Micro Edition），是为电话，机顶盒和P嵌入式电子设备提供的Java语言平台，包括虚拟机和一系列标准化的Java API
3. JavaSE（Java Platform, Standard Edition），用于开发和部署服务器，桌面以及嵌入设备和实时环境中的*Java*应用程序
4. JavaEE（Java Platform，Enterprise Edition）Java EE 是在 Java SE 的基础上构建的，它提供Web 服务、组件模型、管理和通信 API，可以用来实现企业级的面向服务体系结构（service-oriented architecture，SOA）和 Web 2.0应用程序。
   1. 研究目标

在于“启辰资产管理公司”的一段时间接洽中，其提出四大目标，第一为其提供客户自动提示催息的功能；第二提供旗下员工的管理功能；第三能提供报表，数据可视化，该公司希望通过可视化的图表，来直观反映业务的变化；第四，提供订单业务的管理。

* 1. 论文结构

第一章绪论 ：简述Java在开发方面的优势，表述“启辰”公司对项目的四大要求：自动提示催息，员工管理，报表，订单业务。

第二章相关技术与方法：阐述所用到的各种技术，如JavaScript， Html5，ECharts，Spring，MongoDB，BootStrap，jQuery等技术，简述web应用服务器Tomcat以及开发工具Eclipse Neon.1。

第三章系统分析，从业务层次分析了该套系统的各种需求，建立一个方便，直观的系统，实现订单业务，视图，员工管理，客户管理等功能，建立6张数据库表，尽可能的降低耦合度。

第四章系统设计，通过画E-R图设计数据库表，用例图设计出该套系统大致的功能，画顺序图分析功能执行顺序。

第五章系统实现，通过配置数据库事务，简化dao层，着重开发Service层与前端页面，通过前端大量的js脚本去实现强交互性。

第六章系统测试，测试分为两部分，第一部分为开发环境测试，进行功能完整性与压力测试；第二部分进行部署环境测试，测试系统在公网上的体现的效果。

第七章总结，总结近期开发过程出现的问题，与对下一个版本的期望。

* 1. 本章小结

明确了“启辰”公司的需求，确定了开发方向，并确定了项目的大致功能，作为一家以借贷为主要业务的公司，客户的管理和订单业务为该套系统的核心，将采用RIA的开发标准，实现与用户的强交互性，提供优秀的体验。

1. 相关技术与方法
   1. 架构概述

项目将采用MVC模式开发，框架选用Spring,SpringMVC,Spring-data，Bootstrap。

MVC模式：是模型(model)－视图(view)－控制器(controller)的缩写，MVC(Model-View-Controller) 模式基本上算是软件行业的典范，其改变了起初代码难维护，难拓展的的难题，在MVC中，通过将视图层与数据的处理分离，在二次开发中，增加了一定的可拓展性，面对新需求，不用翻动过去的代码，即可实现添加新需求，控制器将只负责页面的调度与数据的返回，model层将进行数据方面的处理，包括从数据库拿取数据，增删改查，封装数据。

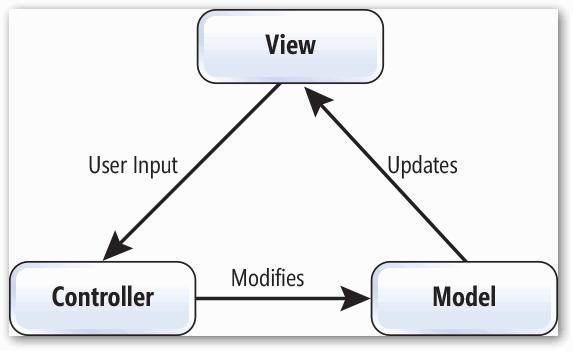


图2.1 MVC

在JavaEE开发中几乎逃不开Spring的和持久化层的框架的使用。

Spring：Spring框架中有许多的组件，其的核心组件只有三个：**Core**、Context和Beans。它们支撑起整个Spring项目。没有了它们就不可能存在有**AOP**、SpringMVC，Spring-data，Spring-boot等上层的特性功能。

Spring框架是一个轻量级的框架，可以一站式地搭建企业级应用。Spring不为取代其他项目，因此它被设计成模块化，他除了支持自己所有的组件外，也对其他框架有着良好的支持，列如：SSH“Spirng+SpirngMVC+Hibernate”，Spring自身也带有对持久层的支持，如Spring-JDBC，但其依旧提供了对叙对优秀持久层框架的支持，因此人们可以自由的选取自己所需的组件进行整合，十分灵活。

Spring是非侵入式的，意味着你的代码通常不会依赖于框架本身。Spring提供注入功能，通过注解，Spring容器将自动给你的代码注入所需类。

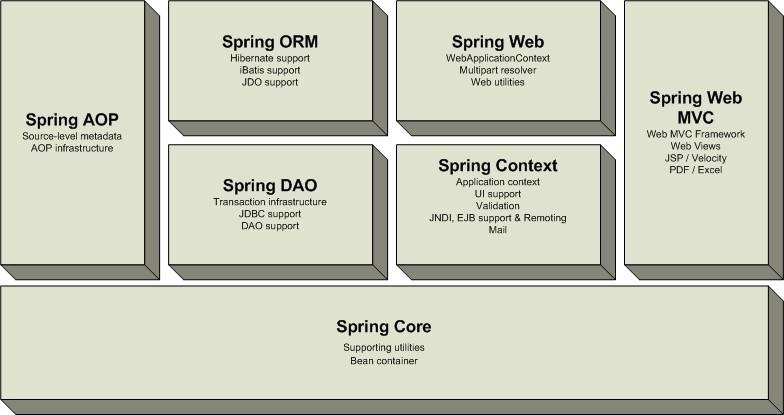


图2.2 Spring框架

Spring MVC：是一种基于Java的请求驱动类型的轻量级Web框架，其优秀的注解能力，以至于其开始取代之前很火热的struts2，成为了当下JavaEE开发的首选Web框架，SpringMVC帮助我们简化开发，仅需进行注解，即可完成对web请求的映射，通过在配置文件中配置DispatcherServlet，即可完成对web请求的拦截，然后转交给处理器映射，映射到对应方法。

**处理器映射**：选择使用哪个控制器来处理请求   
**视图解析器**：选择结果应该如何渲染

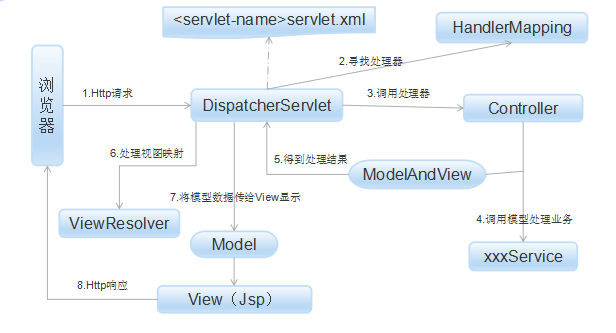


图2.3 SpringMVC

1. JavaBeans，一种特定的规范在Java程序中引入特定的类
2. IoC，控制反转， 简化了开发反复的创建实例的过程，交有Spring容器进行创建并且注入。
3. AOP，Aspect Oriented Programming，面向切面编程
4. POJO，Plain Ordinary Java Object简单的Java对象，实际就是普通JavaBeans，是为了避免和EJB混淆所创造的简称

BootStrap: Bootstrap，是目前很受欢迎的前端框架。Bootstrap是一个简洁灵活的前端CSS/HTML框架，它使得 Web 开发更加快捷。Bootstrap 提供了一个带有网格系统、链接样式、背景的基本结构；Bootstrap 自带以下特性：全局的 CSS 设置、定义基本的 HTML 元素样式、可扩展的 class，以及一个先进的网格系统。

* 1. 关键技术简介

JSP：Java Server Page其根本是一个简化的Servlet设计。是一种动态网页的技术标准。是一种基于html语法的Java拓展，简化了动态页面的开发，使得整个JSP页面的开发，更像是在开发一个Html页面。当下，JSP2.0已经从支持表达语言—EL表达式，它将使得开发人员更加便捷的使用内置对象如session和Java组件等。

JavaScript：JavaScrip是一种web前端的解释性语言，也是一种基于对象（Object）和事件驱动（Event Driven）的、安全性好的脚本语言。它运行在客户端从而减轻服务器的负担。JavaScript主要用来向HTML页面中添加交互行为。JavaScript的用途是解决页面交互和数据交互，最终目的是丰富客户端效果以及数据的有效传递。

Servlet：Servlet是Java为互联网的一种拓展支持，其本身一个接口名，现常指所有处理web请求的类，相对之前的CGI编程，Servlet将更加节省服务器开销，并且速度方面优于CUI。

jQuery：jQuery是一个快速、简洁的JavaScript框架，是Prototype之后又一个优秀的JavaScript代码库（或JavaScript框架）。jQuery中内置了一系列的动画效果，可以开发出非常漂亮的网页。jQuery的选择机制构建于Css的选择器，它提供了快速查询DOM文档中元素的能力，而且大大强化JavaScript中获取页面元素的方式。jQuery可以修改网页中的内容，比如更改网页的样式、插入或者翻转网页图像，jQuery简化了原本使用JavaScript代码需要处理的方式。

ECharts：由百度公司开发，提供了很强的数据可视化能力，其底层以来ZRender，在其官网提供了很多的样例，从柱状图，饼图，到地图，支持多组件混搭

Ajax：Ajax是一种存在很久的技术，并不是一种新的编程语言，通过少量的数据交互，实现局部刷新。对服务器资源有一种节省式的请求，用户不用通过刷新页面才可以获得数据。利用Ajax技术编程，具有很强的交互性。

* 1. 开发工具

该套系统主要在Eclipse Neon.1上进行开发，UML采用StarUML设计，前端切图工具采用photoshop cc 2017，编写页面工具使用Sublime Text;前端使用Twitter开发Boostrap，后台使用Eclipse Neon.1进行开发，Web应用服务器采用Tomcat8.5，数据库采用MongoDB。

* 1. 本章小结

简述了开发该套管理业务系统的部分技术，前端的JQuery与Ajax的配合，与后台进行交互，使得不用刷新页面，也能实时更新数据，百度开发的ECharts提供了可靠的数据可视化方案，MongodDB提供优秀而强大的数据库功能。Spring和Spring MVC是目前JavaEE开发最常用的框架之一，其稳定性已被千万程序员验证过，而Spring-data框架内整合了MongoDB的相关事物，通过XML配置文件，动态的注入MongoTemplate对象，实现对数据库的各种操作，利用这些技术将设计并实现一个符合软件行业的RIA。

1. 系统分析
   1. 可行性分析
      1. 技术可行性

“启辰”公司所提出的四大功能在当前软件开发大背景下，均可实现。其一，实现业绩数据的可视化，将文字的数据转化为直观形象的图表，看似是一门很复杂繁琐的工作。但其实目前已出现了许多优秀的数据可视化工具，比如D3.js，ECharts等，该项目将采用ECharts作为视图化工具，ECharts由百度公司的团队开始，其为开源项目；其二，员工的管理，员工的管理包括了员工的入职离职，工资的统计等，通过数据库的增删改查即可实现该项功能；其三，实现自动提示催息功能，同样也是通过数据库每日的扫描，查询与日期相符的订单记录，然后获取相关客户信息。Spring MVC使得程序更好的处理web请求。四，订单业务管理，通过发布订单请求，匹配订单，完成订单，管理员审批一系列订单流程，实现订单的透明化，更加高效的完成业务。

* + 1. 经济可行性

由于该公司之前采用excel表格来记录业务，不得不需要专门催息，专门制表的相关人员，而人工查询，计算账目，存在过慢，效率低等问题。投入人力物力，却效果繁琐，缓慢。而采用RIA提供强大的查询以及交互性，使得在何时何地都能查看业务，完成业务，只需单次资金的投入，即可让员工专心投入自己的业务中，提高业务能力。优秀的视图的能力，也将帮助公司分析行情。员工手上的客户资源，由于没有一个良好的习惯，去存储客户信息，仅平人的记忆与手机号码，来管理客户，这样存在着客户流失的弊端，于是该套系统也提供了员工对其客户信息的管理 ，方便随时查看。

* + 1. 操作可行性

该套系统上线后，员工通过登陆，可以查看自己月度，季度，年度的业务与工资计算，也可以管理自己的客户，透过数据视图化，能直观看出自己业务能力的变化趋势，公司管理层通过视图第一时间察觉公司业务的变化，而做出相应对策，界面采用美观的背景轮换和柔和的色调，使得员工使用时感觉舒适。通过新建订单的发布，需求匹配订单，完成订单，管理员审批，完成整个业务，可通过订单追述到具体员工，具体人员。其简单性，将有利于员工的操作。

* 1. 需求分析
     1. 系统总体需求

整套系统客户希望，能提供强大的查询功能，页面提供直观的数据呈现力，从图表中发掘行情变化，公司运营方向。对于员工工资的计算要求其具有稳定的正确性；系统的使用是方便，便捷的；同时交互的界面需要具有美观性；对于往后的新需求，系统能提供很好的拓展；同时能兼容各种浏览器。

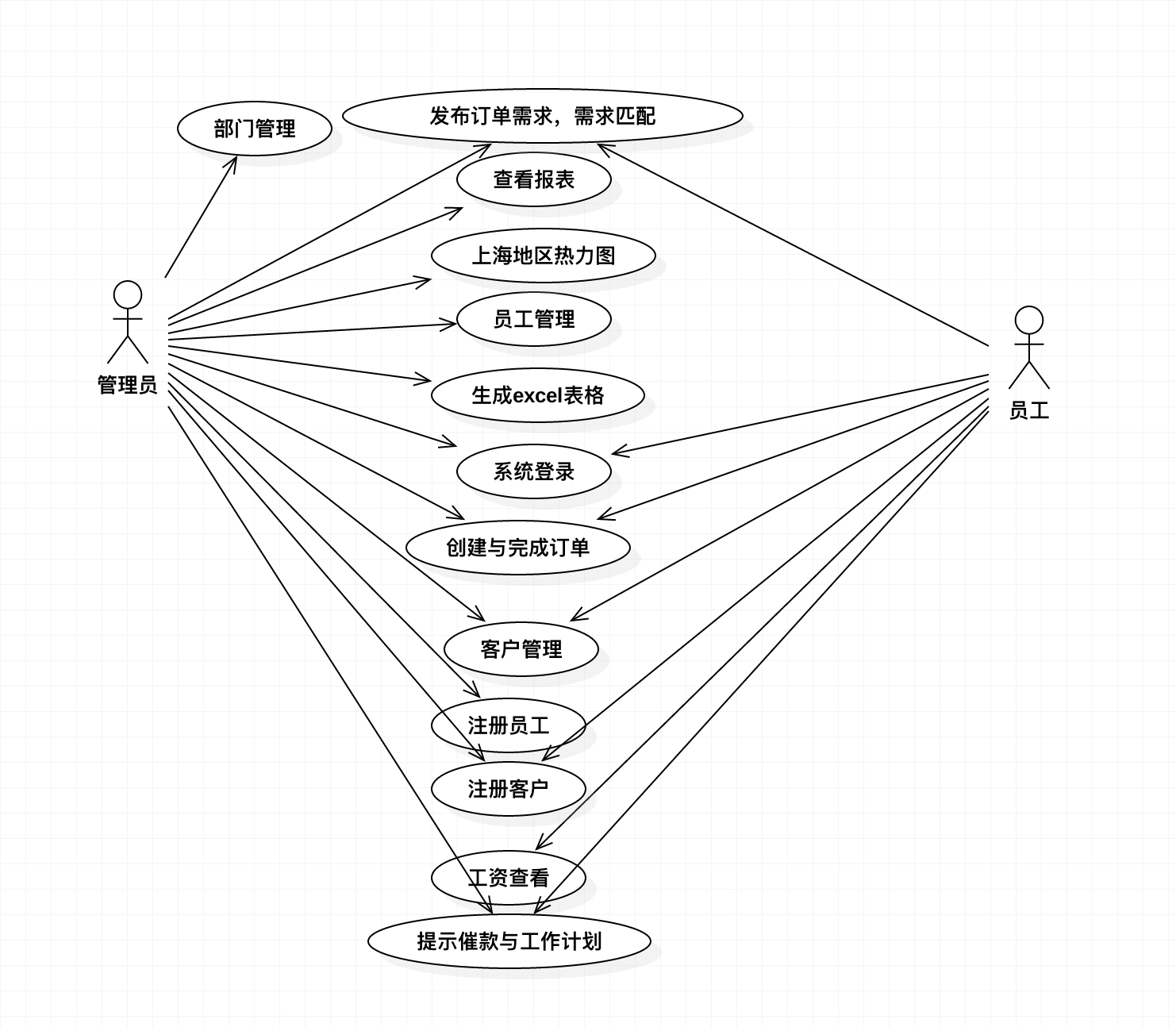
表 3-1 系统需求

|  |  |
| --- | --- |
| 功能需求 | 详细需求 |
| 员工管理 | 对员工的工资进行计算，记录员工个人业绩 |
| 客户管理 | 客户分为两种，投资人与借款客户，对其的个人信息进行管理 |
| 报表 | 每个月生成一份报表，提供参考 |
| 自动提示催息 | 每日定时扫描数据库，获取该星期下应该交息客户 |
| 订单业务 | 创建订单，匹配订单，完成订单，审批，查看订单历史 |

表 3-2 质量要求

|  |  |
| --- | --- |
| 质量属性 | 详细要求 |
| 易用性 | 易用性：软件在使用时，不需要过多复杂的操作 |
| 正确性 | 正确性：在工资，提成等计算上，确保其的正确性 |
| 美观性 | 美观性：能提供一个美观的交互界面 |
| 兼容性 | 员工使用的浏览器五花八门，至少要对主流的浏览器有很好的支持 |
| 可拓展性 | 未来面临新需求时，无须重新开发，仅需二次开发，既能满足新需求 |

* + 1. 用例图分析



系统用例图

系统分为管理端和员工端，管理端整体用例有系统登录、客户管理、查看报表、员工管理，具体说明见表3-2至表3-5：

表3-2 客户管理

|  |  |  |
| --- | --- | --- |
| 用例编号 | UC02 | |
| 用例名称 | 客户管理 | |
| 用例概述 | 管理员通过此用例管理系统中的客户 | |
| 主参与者 | 管理员或员工 | |
| 前置条件 | 管理员或员工身份正确登录 | |
| 基本事件流 | 步骤 | 活动 |
| A1 | 选择需要的操作，增加、修改、删除客户 |
| A2 | 输入客户信息 |
| A3 | 保存输入的信息 |
| 扩展事件流 | 1a | 保存成功 |
| 1b | 用户已存在 |

表3-3 系统登录

|  |  |  |
| --- | --- | --- |
| 用例编号 | UC01 | |
| 用例名称 | 系统登录 | |
| 用例概述 | 管理员通过此用例登录系统 | |
| 主参与者 | 管理员或员工 | |
| 前置条件 | 无 | |
| 基本事件流 | 步骤 | 活动 |
| A1 | 输入正确账户名密码登录 |
| A2 | 进入管理端或员工端 |
| 扩展事件流 | 1a | 用户名不存在或密码错误，返回提示信息 |
| 1b | 成功登录 |

表3-4 员工管理

|  |  |  |
| --- | --- | --- |
| 用例编号 | UC03 | |
| 用例名称 | 员工管理 | |
| 用例概述 | 管理员通过该用例管理员工 | |
| 主要参与者 | 管理员 | |
| 前置条件 | 管理员身份正确登录 | |
| 基本事件流 | 步骤 | 活动 |
| A1 | 选择需要的操作，增加、修改、删除员工，同时核准员工工资 |
| A2 | 输入员工信息 |
| A3 | 保存输入的信息 |
| 拓展事件流 | 1a | 删除成功 |
| 1b | 删除失败 |

表3-5 查看报表

|  |  |  |
| --- | --- | --- |
| 用例编号 | UC04 | |
| 用例名称 | 查看报表 | |
| 用例概述 | 管理员通过该用例查看业绩报表 | |
| 主要参与者 | 管理员 | |
| 前置条件 | 管理员身份正确登录 | |
| 基本事件流 | 步骤 | 活动 |
| A1 | 选择需要操作，查看月份，季度，年度图表 |
| 拓展事件流 | 1a | 数据缺失 |

员工端整体用例有系统登录、客户管理、订单业务、工资查看，具体说明见表3-6至表3-7：

表3-6 查看工资

|  |  |  |
| --- | --- | --- |
| 用例编号 | UC05 | |
| 用例名称 | 查看工资 | |
| 用例概述 | 员工通过该用例查看工资情况 | |
| 主要参与者 | 员工 | |
| 前置条件 | 员工身份正确登录 | |
| 基本事件流 | 步骤 | 活动 |
| A1 | 选择需要操作，查看月份，季度，年度工资情况 |
| 拓展事件流 | 1a | 数据缺失 |

表3-7 订单业务

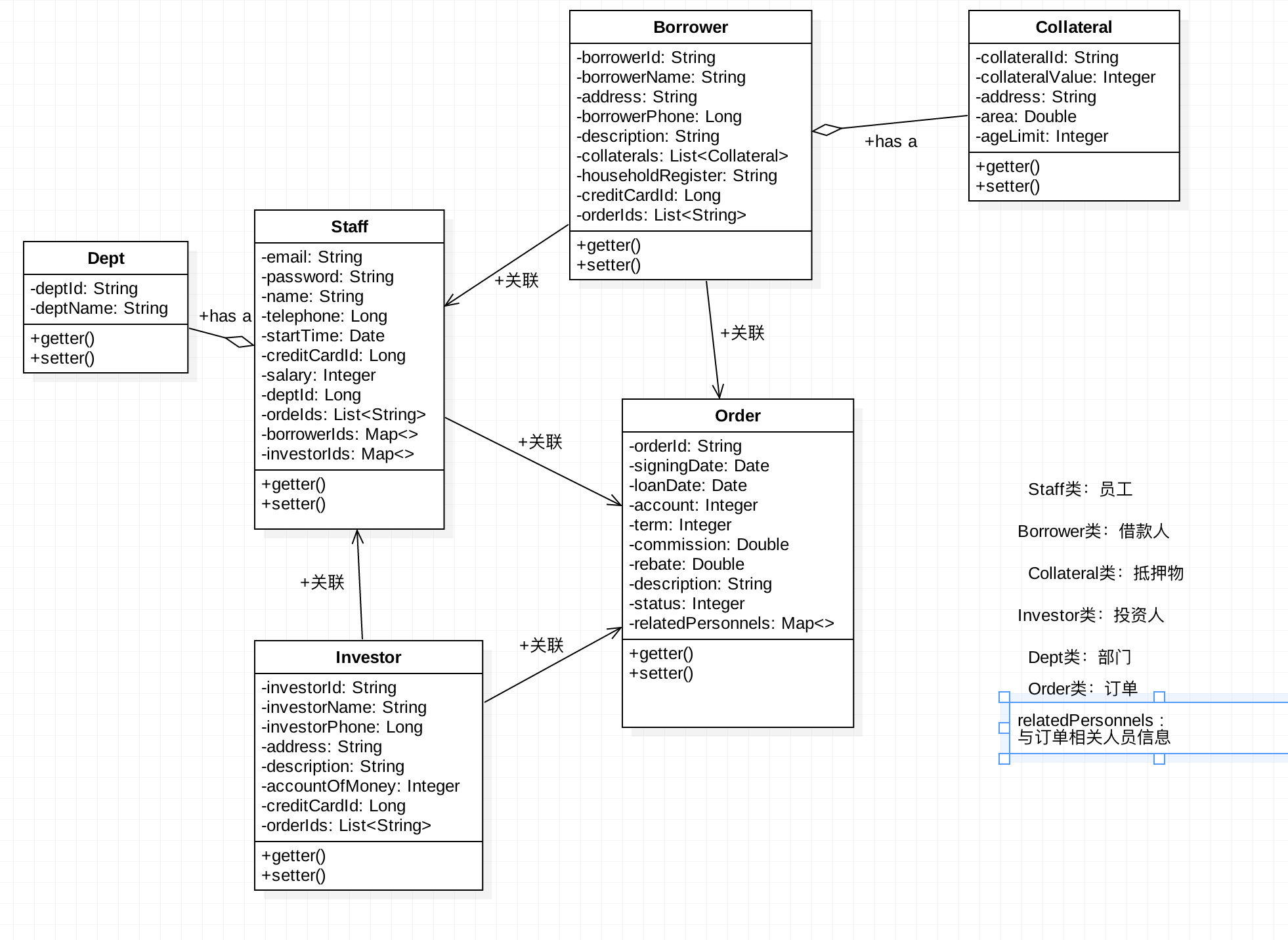
|  |  |  |
| --- | --- | --- |
| 用例编号 | UC06 | |
| 用例名称 | 订单业务 | |
| 用例概述 | 员工通过该用例操作订单 | |
| 主要参与者 | 员工 | |
| 前置条件 | 员工身份正确登录 | |
| 基本事件流 | 步骤 | 活动 |
| A1 | 选择需要操作，修改，删除，提交订单操作 |
| A2 | 待管理员审核通过订单才有效 |
| 拓展事件流 | 1a | 订单完成 |
| 1b | 取消订单 |

* 1. 本章小结

通过用例图分析，明确了整套系统的大致功能，在管理界面，可注册员工，创建订单，记录客户信息，查看历史订单，通过系统登陆模块，区分管理员与员工，员工的订单业务与管理员的报表是整个系统的核心。通过各种数据库的增删改查操作，实现各种管理操作，员工提供客户信息，订单业务，管理员审批订单，查看相关业绩以及发放工资。

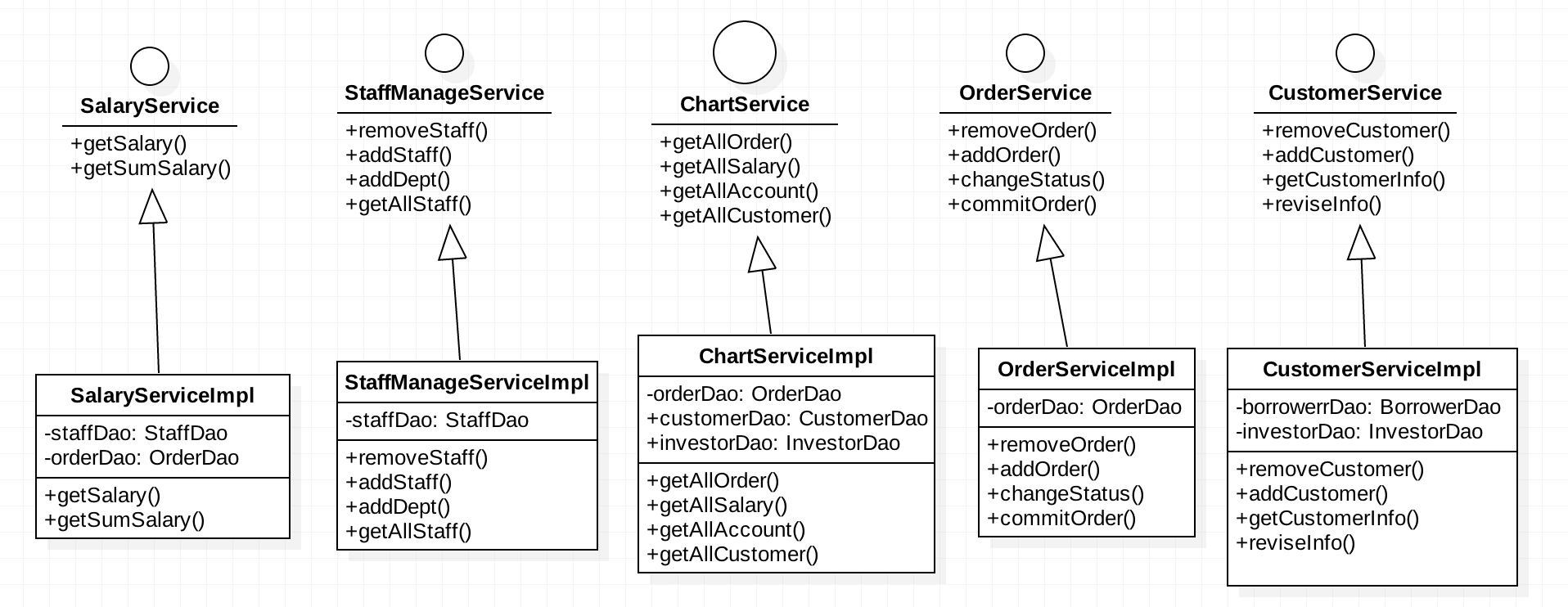
1. 系统设计
   1. 系统类分析

系统基类图：



系统核心bean

该系统核心service见下图：



* + 1. Bean类与Service类

Staff类用于封装员工的相关信息

表4-1 Staff类表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类功能描述 | 封装用户信息 | | | |
| 所在包名称 | com.qichen.po | | | |
| 继承对象 | 无 | | | |
| 实现对象 | 无 | | | |
| 类属性 | | | | |
| 作用域 | 类型 | 名称 | 描述 | 备注 |
| private | String | email | 用户邮箱 | 用户登录账号 |
| private | String | password | 用户密码 | 用户登录密码 |
| private | String | name | 用户名字 |  |
| private | Long | telephone | 用户电话 | 11位数字 |
| private | Date | startTime | 就职时间 | 格式：2017/01/01 |
| private | Date | endTime | 离职时间 | 格式：2017/01/01 |
| private | Long | creditCardId | 银行卡号 |  |
| private | Integer | salary | 工资 |  |
| private | Long | deptId | 部门id |  |

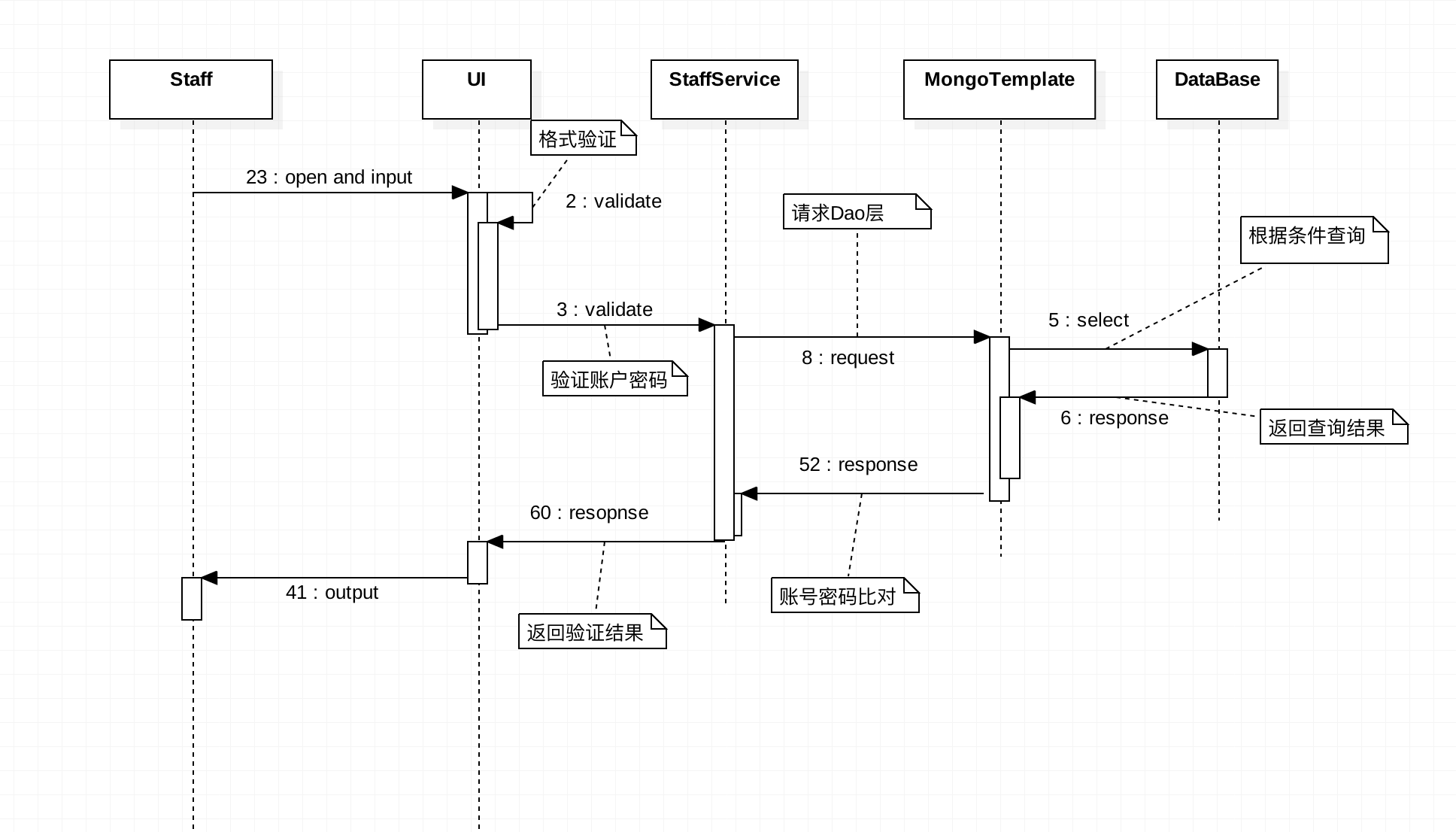
表4-2 StaffManagerServiceImpl类

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类功能描述 | 管理员对员工的增删改查操作 | | | | | |
| 所在包名称 | com.qichen.Service | | | | | |
| 继承对象 | 无 | | | | | |
| 实现对象 | StaffManagerService | | | | | |
| 类属性 | | | | | | |
| 作用域 | 类型 | 名称 | 描述 | | 备注 | |
| private | StaffDao | staffDao | dao层对象 | | 员工dao层对象，请求数据库操作 | |
| 主要实现方法 | | | | | | |
| 作用域 | 方法名 | 参数 | | 返回值类型 | 方法功能描述 |
| public | removeStaff | String email | | void | 通过email主键移除员工 |
| public | addStaff | Staff staff | | void | 将员工添加进数据库 |
| public | addDept | Dept dept | | void | 增加新部门 |
| public | getAllStaff | 无 | | List<Staff> | 查看所有员工 |

StaffServiceImpl实现StaffService接口，StaffService提供了一系列员工管理的方法，例如：添加新员工，移除员工，修改员工所属部门等操作。该服务主要由管理员调用，管理员通过调用实现StaffService的类方法，来实现对员工的操作

表4-3 OrderServiceImpl类

* 1. 关键业务设计
     1. 系统登录



系统登录顺序图

用户通过UI界面输入用户名，密码，通过loginController，StaffService接口继承于StaffManagerService接口，调用StaffService接口的实现类里的获取员工信息方法doValidate()，在doValidate()中将调用StaffDao中的getStaffInfoById()，以查看数据库是否存在相同id的账户，若存在，则返回数据库中的账户，进行账号密码比对，返回对比结果；若数据库中不存在该账号，直接返回错误结果。

* + 1. 注册

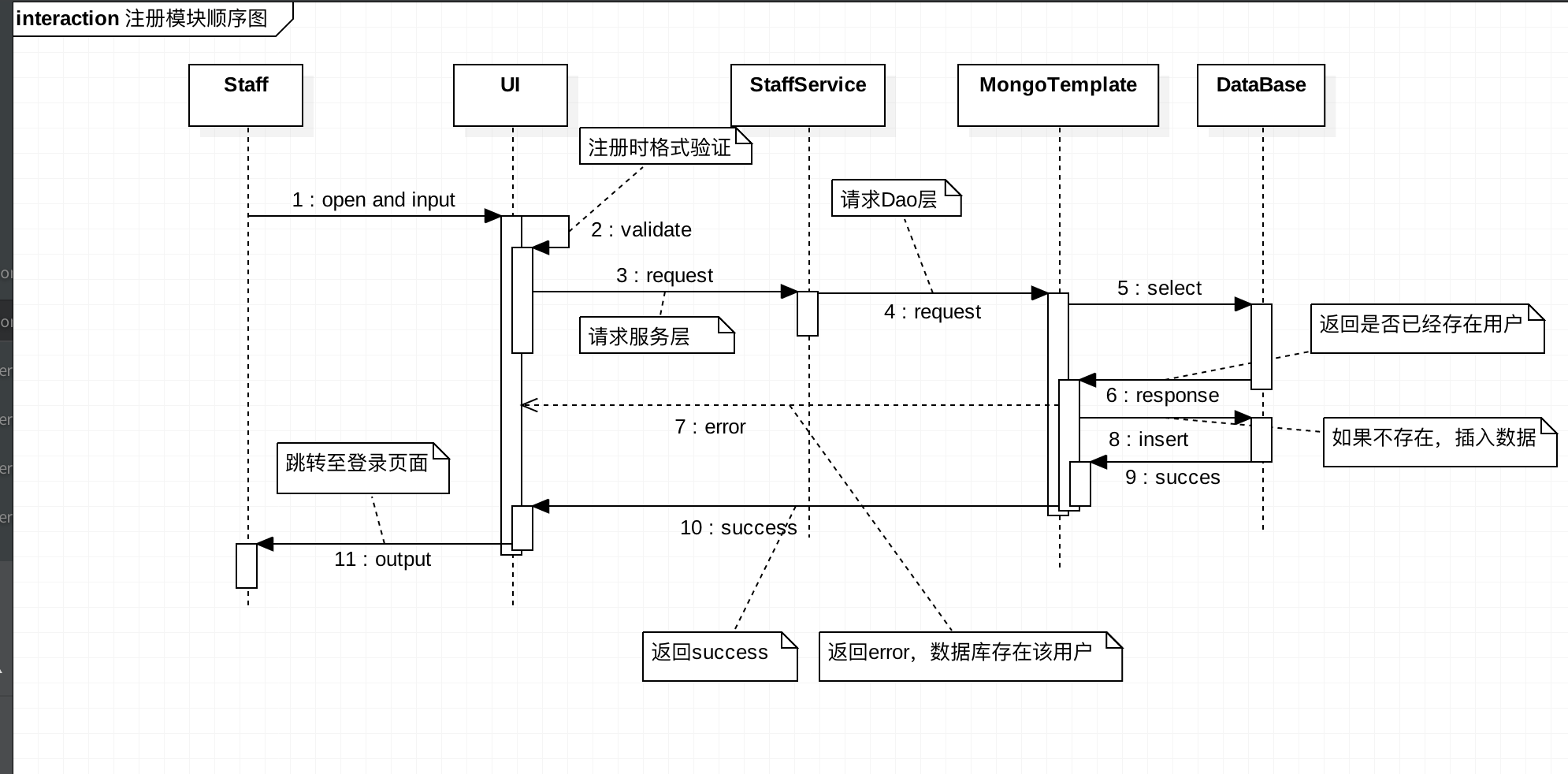


图4-3员工注册模块

管理员通过填写员工信息表单，注册员工信息，在成功注册之前，首先要进行email账号验证，确保数据库中，未存在同样的email，通过一系列的界面提示，帮助成功注册员工。

* + 1. 员工管理

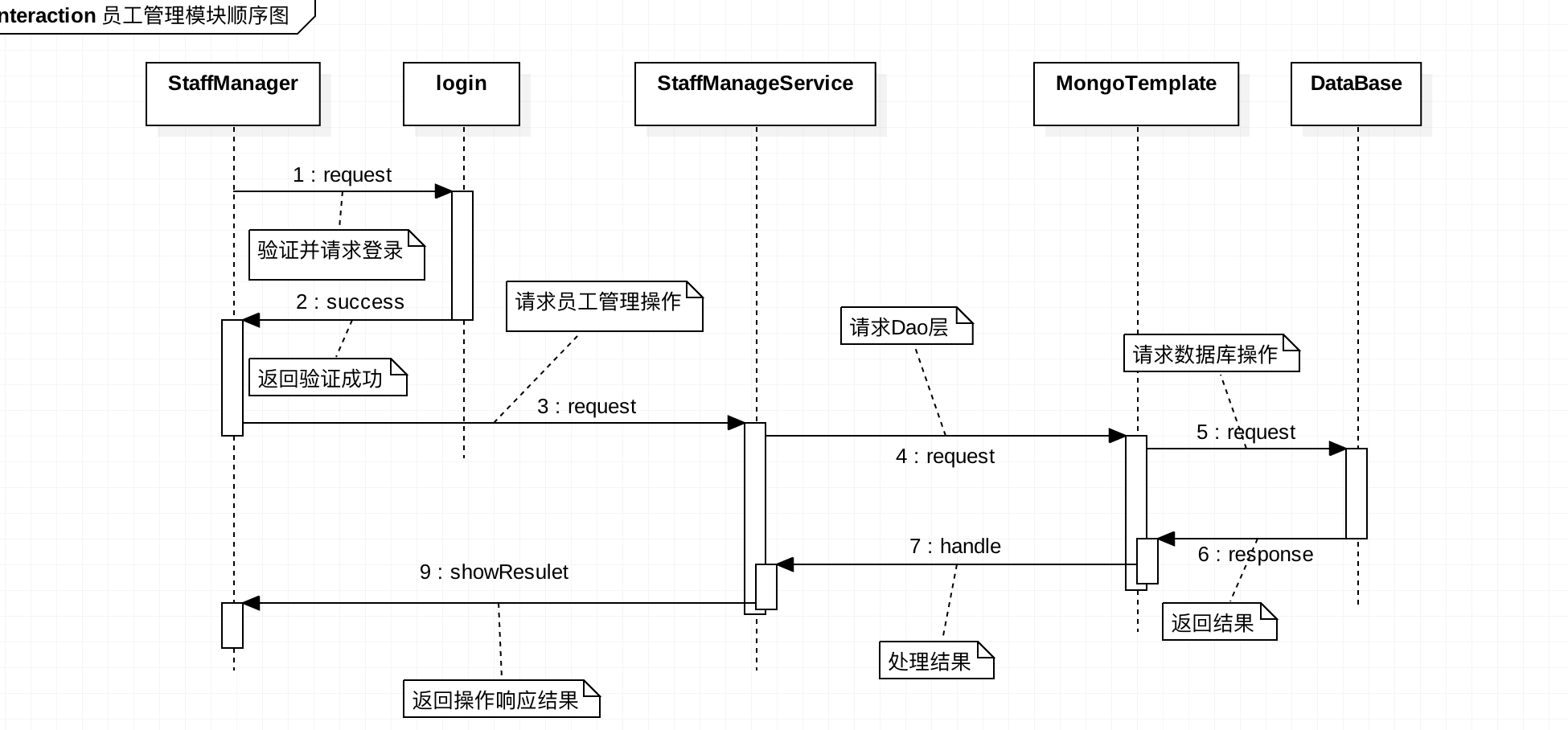


图4-4 员工管理

管理员通过登录，进入员工管理界面，通过各种条件来管理员工，列如部门，工资等等。

* + 1. 客户管理

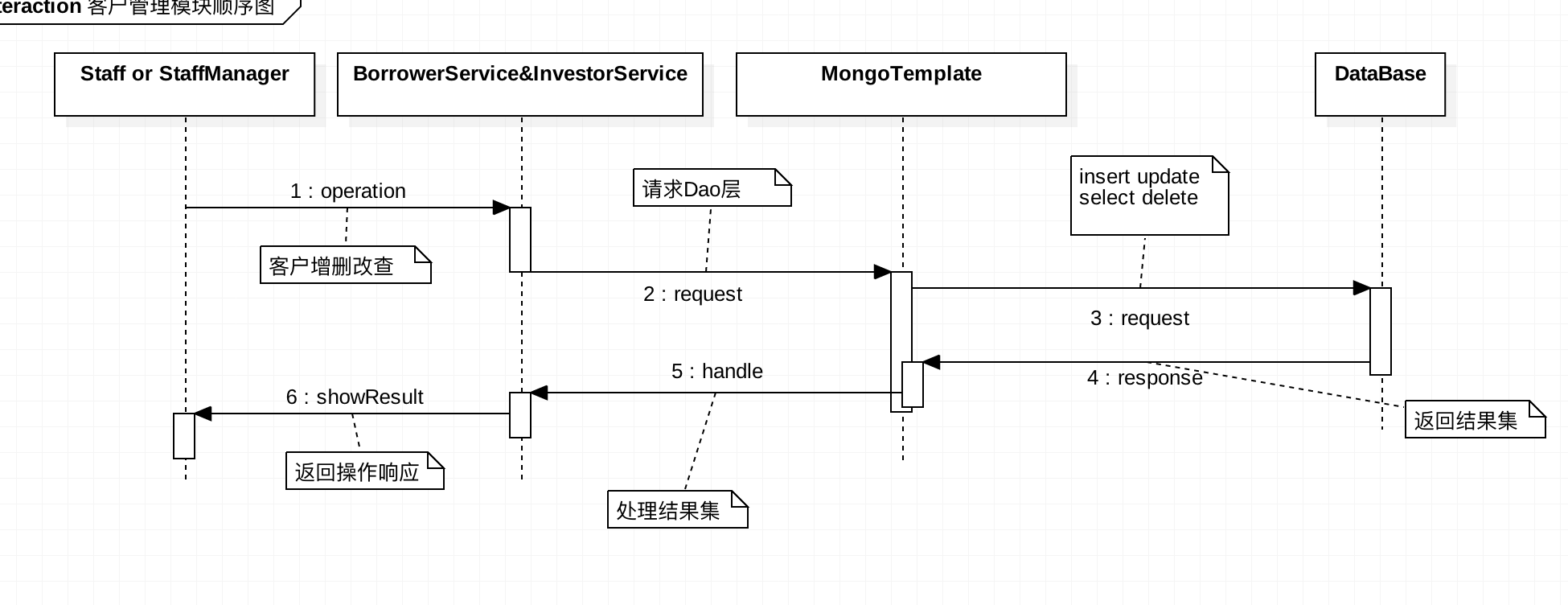


图4-5 客户管理模块

员工或管理员通过调用对应的Service，进行查询自己的客户信息，通过客户的信息，可以进行催款，或约谈等。

* + 1. 查看报表

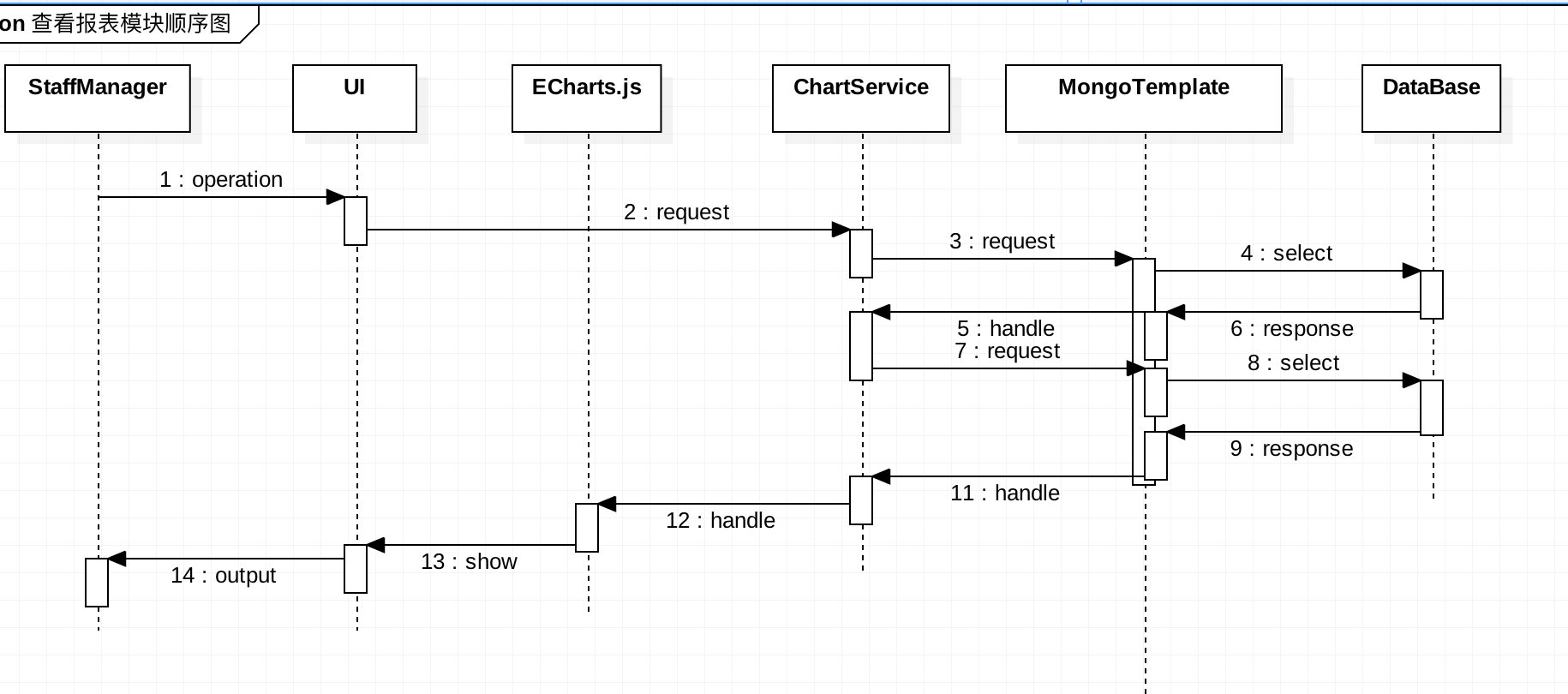


图4-6 查看报表

该功能只有管理员，才可查看。通过权限控制，防止普通员工查看到敏感信息。报表将更加方便，直观的体现公司的业务能力的变化。

* 1. 数据库设计
     1. 概述

数据库将不采用传统的关系型数据库，由于订单业务的复杂型，存在着多个多对多关系，相互关联，传统的关系型数据库不好表达出其复杂的关系，于是选用一种介于关系数据库和非关系数据库之间的产品，MongoDB是非关系数据库当中功能最丰富，最像关系数据库的。他支持的数据结构非常松散。因此可以存储比较复杂的数据类型。Mongo最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立索引。

理论上，可直接将所有数据建立成一张表，出于高耦合的数据表，将降低数据库查询的速度，于是设计成每一个基类Bean，对应着数据库的一张表

* + 1. 概念设计

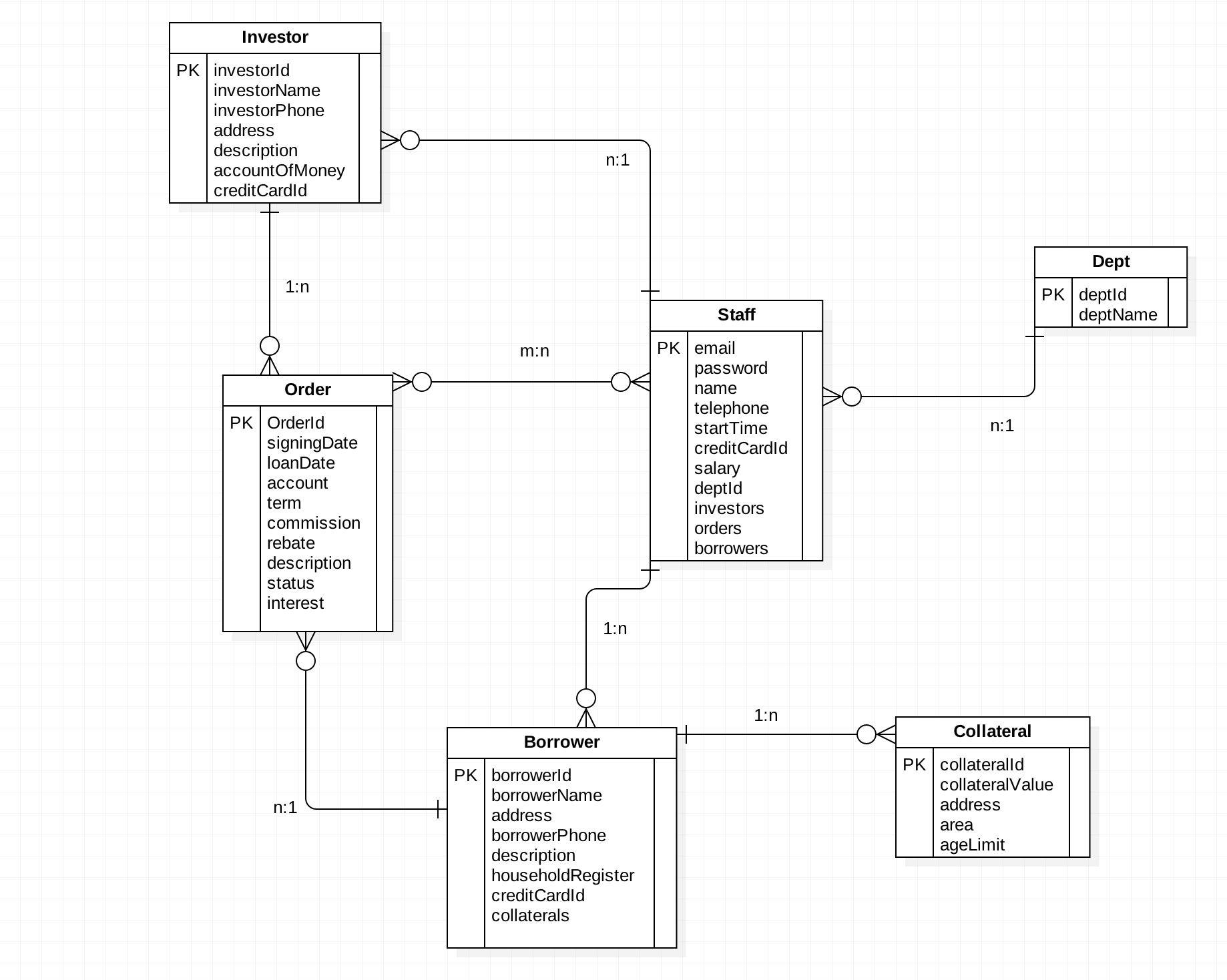


图4-7系统ER图

每一个Investor（投资人）可对应着多个Order（订单），而每个Staff（员工）可对应着多个Investor和多个Order，多个Borrower（借款人），与此同时一个Order可以由多个Staff合作完成，Staff与Dept（部门）为一对一关系，Borrower可以拥有多套抵押物和多个订单。

4.3.3数据库表

系统中使用到数据库基本如表4-4所示。

表4-4系统数据库表

|  |  |  |
| --- | --- | --- |
| 编号 | 表名 | 描述 |
| 01 | Staff | 员工表 |
| 02 | Order | 订单表 |
| 03 | Borrower | 借款人表 |
| 04 | Investor | 投资人表 |
| 05 | Dept | 部门表 |
| 06 | Collateral | 抵押物表 |

数据库中的用户表用来存储各类用户的相关数据，每个用户都要在用户表中有相相应的记录。日志表用于记录每个用户的基本操作。

1. 用户表

系统用户表用于登录系统使用，用户表的具体字段如表4-2所示。

表4-5员工表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 类型/长度 | 约束 | 备注 |
| email | 员工账号 | String | PK | 邮箱作为登录帐号 |
| name | 员工名称 | String | NOT NULL | @Indexed 添加索引 |
| password | 员工密码 | String | NOT NULL |  |
| telephone | 电话号码 | Long |  |  |
| starttime | 入职时间 | Date |  | 注册时自动添加 |
| salary | 工资 | String |  |  |
| investors | 投资人映射 | Map |  | Map存储映射快速查询 |
| borrowers | 借款人映射 | Map |  | Map存储映射快速插叙 |
| orders | 订单列表 | List |  | List存放订单主键 |
| deptId | 部门id | Long |  | 员工所在部门 |

字段员工账号为本表的主键用来标识一个用户，员工帐号和密码是两个非空字段，用户使用用户名和密码作为登录系统的凭证，通过部门id来区分管理员与普通员工，两个Map映射，将提供更快的查询速度，查看相关信息。

（2）订单表

表4-6 Order表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段描述 | 类型／长度 | 约束 | 备注 |
| orderid | 订单号 | String | PK | 主键唯一标识订单 |
| signingDate | 签约日期 | Date | Not null |  |
| loanDate | 放款日期 | Date | Not null |  |
| account | 借款金额 | Integer | Not null |  |
| term | 期限 | Integer | Not null |  |
| commission | 总佣金 | Double | Not null |  |
| interest | 利息 | Double | Not null |  |
| rebate | 返点 | Double |  | 与外界公司合作，存在返点 |
| description | 订单描述 | String |  |  |
| status | 订单状态 | Integer | Not null | 状态检测，判定订单是否有限 |

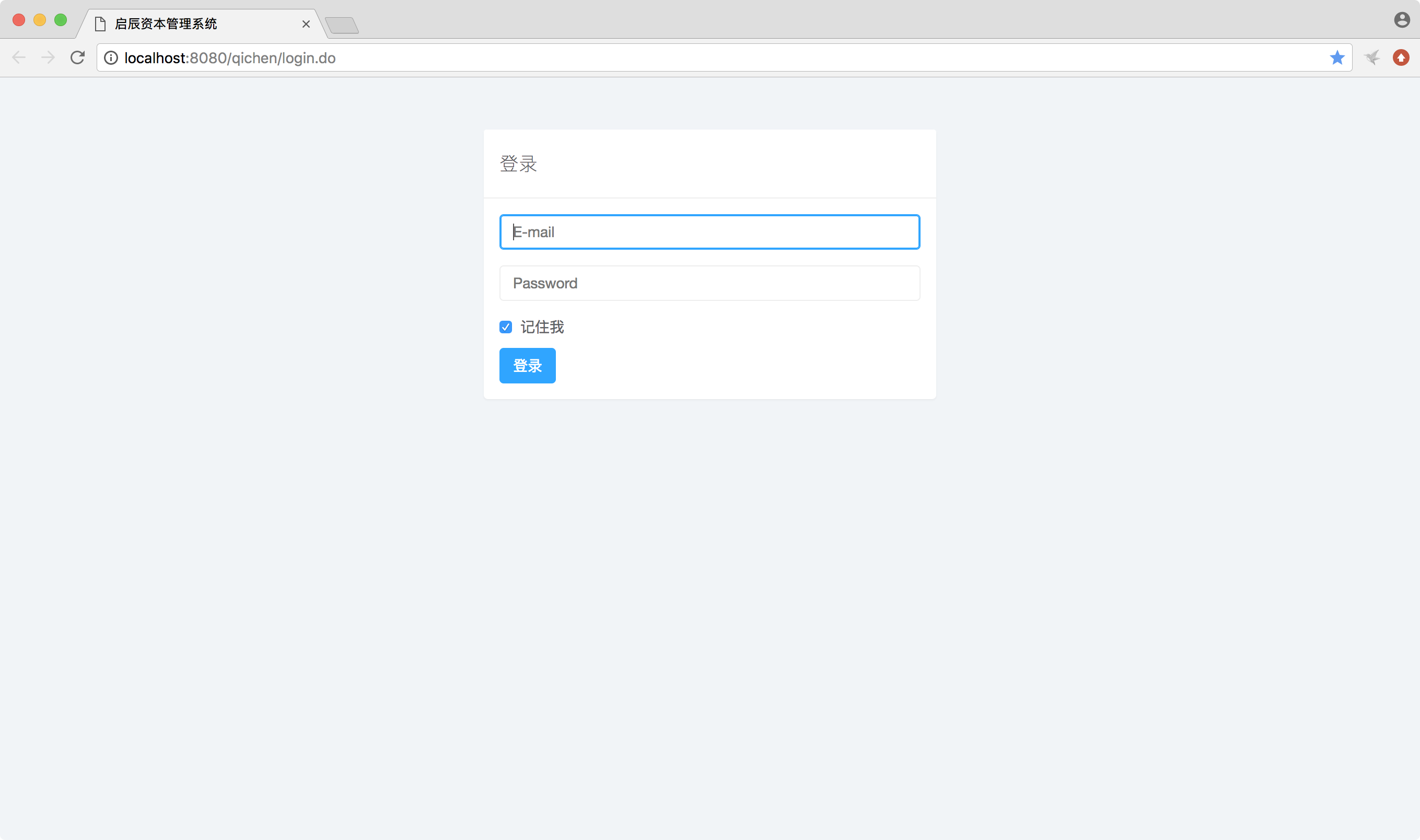
Order表对应着整个公司的业务,客户通过签约实现资金周转，因此签约日期需被设计成非空，而在总佣金，返点这两项数据，存在着两种计算方式，一种直接金额，另一种取比例，因为需要一种通过的数据结构，根据Java的数据类型精度向上原则，因此取比例的double类型。返点只有与外界公司合作时存在，通过控制订单状态，来判断订单所属流程，判断订单是否有效。

* 1. 本章小结

通过画E-R图，将各个数据表之间的关系，清晰的呈现出来，一个订单与员工的多对多关系，员工与客户的一对多关系。客户，订单都通过Staff（员工）联系在一起，这也很符合现实情况，借款人与投资人都是通过员工方可达成订单，完成订单。员工将很方便快捷的查询其想查询的内容。

1. 系统实现
   1. 模块实现
      1. 员工登陆
2. 功能描述

登陆模块采用简约的风格，通过员工登录，区分是否为管理员，从而加载不同模块。通过登录后，员工将跳转至home.jsp



员工登录

1. 实现流程

在登录页面填写账号与密码，格式将由插件bootstrapvalidator.js，进行验证，之后通过登录按钮，向后台doLogin.do发送表单数据，进行账号与密码匹配，调用validateStaff（）方法，若返回true 则为验证通过，随后调用getStaffInfo（）方法获取员工数据，提供给home.jsp。若返回false，返回一个密码或账号错误的弹框提示。

* + 1. 员工注册

(1)功能描述

员工注册模块，通过一系列的时间触发，进行表单验证，当email输入框，失去焦点时，触发验证email是否存在，通过ajax与后台数据库进行验证。且每个输入框，均绑定焦点事件，确保提交的注册信息不存在空，同时bootstrapvalidator将对数据格式进行验证。最后通过点击注册按钮，用ajax发送数据给后台。

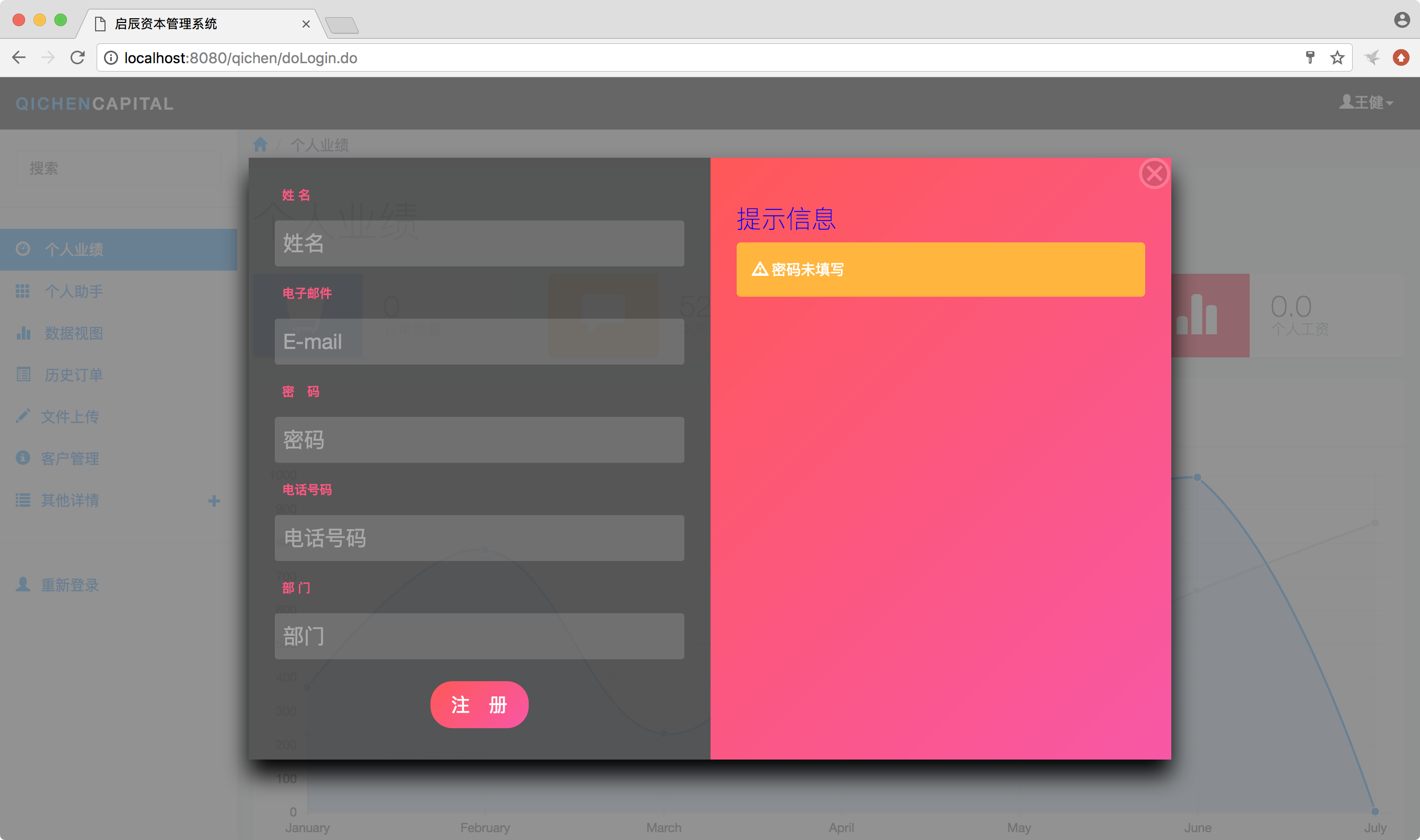


图5-2员工注册

(2)实现流程

通过将焦点事件与输入框绑定，进行实时验证与提示，帮助注册员工，其中email需要通过ajax向后台发送数据，进行验证，确保email的唯一，而数据格式的规整，插件bootstrapvalidator将帮助验证。成功注册后，将执行一个30秒的倒计时，提示是否重新登录

(3)实现代码

<div id="mask"> <!—遮盖层-->

<div class="modal hidden" style="position:fixed;">

<div class="form" ><!—注册表单 -->

    <form id="register">

      <label class="label" for="name" required="required">姓 名</label>

      <input id="name" class="input" type="text" placeholder="姓名"/>

      <label class="label" for="email" required="required">电子邮件</label>

      <input id="email" class="input" type="email" placeholder="E-mail"/>

      <label class="label" for="pass">密　码</label>

      <input id="pass" class="input" type="password" required="required"  placeholder="密码"/>

      <label class="label" for="telephone">电话号码</label>

      <input id="telephone" class="input" type="text" placeholder="电话号码"/>

      <label class="label" for="dept"  required="required">部 门</label>

      <input id="dept" class="input" type="text" placeholder="部门"/>

      <button type="button" class='button'>注　册</button>

    </form>

  </div>

  <div class="invite">

    <h3 style="color:blue;">提示信息</h3><div id="Msg""></div>

    <div class="nope1" style="color:yellow;display:none;" >取消</div>

    <div class="nope" style="color:yellow;display:none;"  >确定</div>

    <div title="close" class="close"></div>

</div>

鼠标焦点事件

$("#name").blur(function(){

console.log($("#name").val());

var div1 ='<div class="alert bg-warning" role="alert"><span class="glyphicon glyphicon-warning-sign"> </span>  姓名未填写</div>'

if($("#name").val()==""||$("#name").val()==undefined){

$("#Msg").html(div1);

}

});

$("#pass").blur(function(){

var div1 ='<div class="alert bg-warning" role="alert"><span class="glyphicon glyphicon-warning-sign"> </span>  密码未填写 </div>'

if($("#pass").val()==""||$("#pass").val()==undefined){

$("#Msg").html(div1);

}

});

$("#dept").blur(function(){

var div1 ='<div class="alert bg-warning" role="alert"><span class="glyphicon glyphicon-warning-sign"></span>  部门未填写 </div>'

if($("#dept").val()==""||$("#dept").val()==undefined){

$("#Msg").html(div1);

}

});

//email 验证

$("#email").blur(function(){

var div1 ='<div class="alert bg-warning" role="alert"><span class="glyphicon glyphicon-warning-sign"></span>  email未填写 </div>'

if($("#email").val()==""||$("#email").val()==undefined){

$("#Msg").html(div1);

return ;

}

$.ajax({

type:"POST",

url:"<%=contextPath%>/doValidateEmail.do",

dataType:"json",

data:"email="+$("#email").val(),

success:function(data,status){

if(status == "success"){

if(data.result == true){

var div ='<div class="alert bg-success" role="alert">'

+'<span class="glyphicon glyphicon-check"></span>  email 验证成功！请继续完成注册</div>';

$("#Msg").html(div);

              }else{

              var div='<div class="alert bg-danger" role="alert">'

  +'<span class="glyphicon glyphicon-exclamation-sign"></span>  email 已被注册 请选择其他email进行注册</div>'

              $("#Msg").html(div);

              }

}

},

error : function(data) {

var div='<div class="alert bg-danger" role="alert">'

  +'<span class="glyphicon glyphicon-exclamation-sign"></span>   Interent Error </div>'

              $("#Msg").html(div);

console.log(data.responseText);

        }

});

});

$(".button").on("click",function(){

var div1 ='<div class="alert bg-warning" role="alert"><span class="glyphicon glyphicon-warning-sign"> </span>   注册信息未填写完全 </div>'

if($("#email").val()==""||$("#name").val()==""||$("#pass").val()==""||$("#telephone").val()==""||$("#dept").val()==""){

$("#Msg").html(div1);

return;

}

$(".button").attr("disabled",true);

$.ajax({

type:"POST",

url:"<%=contextPath%>/doRegisterStaff.do",

dataType:"json",

data:"&email="+$("#email").val()+"&name="+$("#name").val()+"&password="+$("#pass").val()+"&telephone="+$("#telephone").val()+"&dept="+$("#dept").val(),

success:function(data,status){

if(status == "success"){

if(data.result == true){

var div ='<div class="alert bg-success" role="alert">'

+'<span class="glyphicon glyphicon-check"></span>  新员工注册成功！请在<strong id="rankTime" style="font-size:26px;color:red;">30</strong>秒内请选择是否切换用户 </div>';

$("#Msg").html(div);

/\* var html ='<strong id="rankTime" style="font-size:30;color:red;">'+30+"</strong>";

$(".alert").append(html); \*/

$(".nope").css("display","block");

$(".nope1").css("display","block");

t=setInterval("settime()",1000);

}else{

var div='<div class="alert bg-danger" role="alert">'

  +'<span class="glyphicon glyphicon-exclamation-sign"></span>  新员工注册失败，请重新注册</div>'

              $("#Msg").html(div);

}

}

},

error : function(data) {

var div='<div class="alert bg-danger" role="alert">'

  +'<span class="glyphicon glyphicon-exclamation-sign"></span>  新员工注册失败，请重新注册</div>'

              $("#Msg").html(div);

console.log(data.responseText);

        }

});

$(".button").removeAttr("disabled");

});

//取消

$(".nope1").on("click",function(){

$(':input','#register')

.not(':button, :submit, :reset, :hidden')

.val('')

.removeAttr('checked')

.removeAttr('selected');

$("#Msg").html("");

$("#rankTime").html("");

clearInterval(t);

time=30;

$(".button").removeAttr("disabled");

$("#mask").hide();

$("body").css("overflow","scroll");

});

//确定

$(".nope").on("click",function(){

window.location="<%=contextPath%>/login.do";

});

后台相应代码

@Resource

private StaffService staffServiceImpl;

@Resource

private BorrowerService borrowerServiceImpl;

@Resource

private InvestorService investorServiceImpl;

@Resource

private OrderService orderServiceImpl;

/\*\*

\* 邮箱验证controller

\* @param email

\* @return

\*/

@RequestMapping("doValidateEmail.do")

@ResponseBody

public JSONObject ValidateEmail(String email){

if(log.isInfoEnabled()){

log.info("Enter method ValidateEmail email :"+email);

}

Staff staff = staffServiceImpl.getStaffInfo(email);

JSONObject jsonObject = new JSONObject();

if (staff==null) {

jsonObject.put("result", true);

if(log.isInfoEnabled()){

log.info("Leave method ValidateEmail result : true");

}

}else {

jsonObject.put("result", false);

if(log.isInfoEnabled()){

log.info("Leave method ValidateEmail result : false");

}

}

return jsonObject;

}

/\*\*

\* 注册员工controller

\* @param email

\* @param name

\* @param telephone

\* @param password

\* @param dept

\* @return

\*/

@RequestMapping("doRegisterStaff.do")

@ResponseBody

public JSONObject registerStaff(String email,String name,String telephone,String password,String dept){

if(log.isInfoEnabled()){

log.info("Enter method doRegisterStaff email :"+email);

}

Staff staff = new Staff(email, password, name, Long.valueOf(telephone), new Date(), null, null, null, null, 1000L);

staffServiceImpl.addStaff(staff);

JSONObject jsonObject = new JSONObject();

jsonObject.put("result", true);

if(log.isInfoEnabled()){

log.info("Leave method doRegisterStaff !");

}

return jsonObject;

}

* 1. 本章小结

通过触发前端页面的事件，与后台进行交互，因此需要大量的JavaScript代码来满足交互需求，通过ajax发送数据给后台，后台接收，调用对于的Service，Service调用MongoTemplate内的方法，如insert（），upsert（）等等，以进行数据库各种操作 。整套系统的代码主要集中在前端页面，后台由于采用mongoDB数据库，Spring-data 提供了数据库事务操作，因此dao层将不存在。

1. 系统测试
   1. 系统测试综述

系统将分为两个阶段，第一阶段：本地开发环境测试，测试环境为Mac OS X，部署容器为tomcat8.5，JDK版本为1.7。首先进行功能测试，测试功能的完整性与稳定性，其次将测试该系统的极限压力能力，列如：数据过多时，页面样式控制等。第二阶段：发布环境测试，测试环境为阿里云的ECS，系统为Linux，一核2G，JDK1.7。同样进行稳定性与压力测试。

* 1. 测试用例
     1. 员工登录模块

员工登录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能测试 | | | | | |
| 概述 | | | | | |
| 测试编号 | | | QC001 | | |
| 功能描述 | | | 员工登录 | | |
| 用例目的 | | | 测试用户是否能成功登录登出 | | |
| 前提条件 | | | 进入用户登录界面 | | |
| 测试操作 | | | | | |
| 编号 | 输入 | 期望的响应 | | 实际情况 | 是否正确 |
| 1 | 输入不存在的用户名，不填写密码 | 系统提示用户名或密码错误 | | 系统提示登录名或密码错误 | 正确 |
| 2 | 输入正确的用户名输入错误的密码 | 系统提示用户名或密码错误 | | 系统提示登录名或密码错误 | 正确 |
| 3 | 输入错误的用户名输入正确的密码 | 系统提示用户名或密码错误 | | 系统提示登录名或密码错误 | 正确 |
| 4 | 不填写用户名及密码 | 系统提示用户名或密码错误 | | 系统提示用户名或密码错误 | 正确 |

* + 1. 发送消息模块

表6-2发送消息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能测试 | | | | | |
| 概述 | | | | | |
| 测试编号 | | | QC002 | | |
| 功能描述 | | | 发送离线消息 | | |
| 用例目的 | | | 测试消息是否成功发送，被查看 | | |
| 前提条件 | | | 进入个人助手界面 | | |
| 测试操作 | | | | | |
| 编号 | 输入 | 期望的响应 | | 实际情况 | 是否正确 |
| 1 | 发送无标题消息 | 显示弹框发送成功 | | 显示成功 | 正确 |
| 2 | 发送给无效员工账号 | 显示错误提示是非法地址 | | 成功提示，并且不允许发送 | 正确 |
| 3 | 查看消息时，用完即毁 | 查看完未读消息，即数据库删除消息 | | 后台自动删除 | 正确 |
| 4 | 点击未读消息，如为0条 | 未读消息数为0时，弹框提示 | | 成功提示 | 正确 |

* 1. 测试分析

对测试的结果是否达到预期进行分析，未达到预期的给出改进方法。

在生产环境的测试结果与部署环境近乎一致，交互效果良好，压力测试时，存在少量页面样式问题，由于个别div的width与height未给定固定值，超过长度，会影响整体样式。功能测试暂无任何错误。

* 1. 本章小结

一个优秀的软件系统，除了优良的设计与实现之外，仍需一套完整的测试，方可部署上线，通过系统的测试，发现问题，优化问题。这其中不仅仅是解决问题了，同时也对设计的一种优化，弥补前期设计上的一些考虑不周到。在压力测试中，要测试在各种极端环境下，软件是否在正常运行，列如：并发访问时，数据库操作的稳定性。大量数据时，页面样式的显示问题。

1. 总结

整个毕业设计，花费了将近一个月的时间，历经两个版本的改动，中途更换整体设计页面样式，尽管过程时曲折的，但是最终测试的结果还是让人欣喜的，项目也将成功，为启辰资产管理公司提供一套完整的服务，日后将会持续对该项目进行优化，项目在代码设计上，存在一些缺点，脚本代码过于臃肿，一些公共部分，并为抽离出来。这些问题将在2.2版本的项目中，进行逐一优化，2.2版本中将加入目前流行的kafka与redis，作为消息服务更好支持。通过该系统的实现，充分应用了自己在实习过程中所学到的知识，从最开始的模仿，到现在的自己独立开发，这中间的过程是美好的。整套系统的开发是艰辛的，由于与工作之间的冲突，导致项目的迟缓与设计上的不足。

附页 小论文

摘 要：随着互联网与IT技术的迅猛发展，基于网络的web应用软件成为当前和未来的软件系统开发的主流。传统的低交互性程序已无法满足社会的需求，为了提供web应用程序具有更强的交互性，将采用大量的web前端技术，本文在大量的RIA开发经验基础上，设计并开发某资产管理公司的员工管理与业务系统，采用B／S结构（Browser/Server），一个优秀的RIA是离不开脚本语言的支持，通过ajax技术与后台进行异步交互，JavaScript语言是目前RIA开发的标准。而数据库采用MongoDB，一个介于关系型数据库与非关系型数据库的产品。

关键词：Ajax，RIA，JavaScript

Abstract: With the rapid development of Internet and IT technology, network-based web applications become the mainstream of current and future software system development. The demand of the society has been unable to meet the low interactivity of traditional program, in order to provide web application has a stronger interaction, will use a large number of Web front-end technology, based on the basis of a lot of experience in RIA development, staff management and service system design and development of a Asset Management Co, the B / S structure (Browser/Server), a good RIA cannot do without the support of script language, asynchronous interaction using Ajax technology and background. The JavaScript language is the current standard of RIA development. The database uses MongoDB, a relational database and non relational database products.

Keywords: Ajax, RIA, JavaScript

绪论：

1.研究背景和意义：“上海启辰资产管理公司”是一家资产借贷公司，提供给客户的优秀的资产管理。由于其客户群愈发增多，传统的excel表格检索已无法满足其的需求，excel表格也无法提供每日提示催息的功能，无法直观的显示其业绩。翻看中外的开发书籍，采用Java语言开发无疑是最好的选择。Java语言是目前最流行的高级编程语言。

2.研究目标：在于“启辰资产管理公司”的一段时间接洽中，其提出四大目标，第一为其提供客户自动提示催息的功能；第二提供旗下员工的管理功能；第三能提供报表，数据可视化，该公司希望通过可视化的图表，来直观反映业务的变化；第四，提供订单业务的管理。

相关技术与方法：

1.架构概述：项目将采用MVC模式开发，框架选用Spring，SpringMVC Spring-data，Bootstrap。

2.关键技术简介：JSP：Java Server Page其根本是一个简化的Servlet设计。是一种动态网页的技术标准。JavaScript：JavaScrip是一种web前端的解释性语言，也是一种基于对象（Object）和事件驱动（Event Driven）的、安全性好的脚本语言。jQuery：jQuery是一个快速、简洁的JavaScript框架，是Prototype之后又一个优秀的JavaScript代码库（或JavaScript框架）。

系统设计：

由于整体系统采用的是MonggoDB，该数据库可直接存储Java对象，以json对象的二进制形式进行存储，因为不用考虑是否建立类与类的关系。前端框架采用bootstrap，其拥有很多的插件，列如：bootbox——提示框，datepicker——日期选择等等。

功能设计方面，该套系统拥有一套完整的功能，从安排工作计划，到公司业务的创建与完成等等一系列功能。在个人助手页面，将提供修改密码，注册员工，发送消息，查看未完成订单，查看消息，添加工作安排，修改订单，完成订单，提交订单等功能；在客户管理界面，将提供查看客户列表，客户详情；在数据视图界面上，将提供查看一系列图表。

系统实现：

在添加工作计划模块中，选用了datepicker插件，触发时间选择事件后，显示工作计划表单，通过ajax提交保存。在发送消息模块中，email的填写，会触发焦点事件，通过ajax与数据库进行比对email账号是否存在且账号不能为自己，不存在则无法发送消息。在数据视图模块，通过百度公司的echarts插件，会呈现一个上海地区的热力图，炫目的效果，但同时出体现客户的热点地区。

系统测试：

测试将分为2部分，一部分为生产环境的测试，另一部位为部署环境的测试。生产环境为Mac os，部署环境为阿里云ECS。首先进行功能完整性测试，然后进行压力测试。

总结：独立开发一套完整的系统，是一个有趣且辛苦的过程，验证了许多新技术，通过提高了自身独立开发能力，整体项目符合预期效果，还有部分高级功能，将在2.2版本中加以实现，列如：加入solar搜索引擎，对全文进行检索，加入消息队列kafka的支持。

# 参考文献

[1]Bruce Eckel.《Thinking in Java》．北京：机械工业出版社，2007

[2] Gary Cornell.《JAVA核心技术》.北京:电子工业出版社, 2011

[3] 霍多罗夫.《MongoDB权威指南》.北京：人民邮电出版社,2011

[4]徐涛.《深入理解Bootstrap》北京：.机械工业出版社,2014

[5]《jQuery教程》. W3School

[6]Nicholas C.Zakas. 《JavaScript高级程序设计》.北京.人民邮电出版社，2006

[7]巴萨姆.《Head First Servlet JSP》.中国电力出版社.2006

[8]Paul Deck.《SpringMVC学习指南》.北京.人民邮电出版社.2015

[9]百度. 《ECharts》.百度公司

[10]程晓明.《深入理解Java内存模型》

[11]周志明. 《Java虚拟机高级特性与最佳实践》.北京.机械工业出版社

[12]布雷登巴赫.《Spring in Action》.北京.人民邮电出版社.2008

[13]极客学院.《Tomcat8权威指南》.2015

[14]Joshua Bloch.《Effective Java》.北京.机械工业出版社.2003

[15]鸟哥.《鸟哥的Linux私房菜》北京.人民邮电出版社.2016

# 致 谢

四年的学习生活在即将划上一个句号，而于我的人生来说却仅仅只是一个逗号，我将面对新的征程的开始。本研究及论文实在我的导师柴政的亲切关怀和耐心的指导下完成的。论文文章的完成，每走一步对我来说都是新的尝试，这也是我在大学期间独立完成的最大的项目。在这段时间里，我学到了很多知识也有很多感受，从一无所知，我开始了独立的学习和试验，查看相关的资料和书籍，让自己头脑中模糊的概念逐渐清晰，使自己非常稚嫩作品一步步完善起来，每一次改进都是我学习的收获，每一次功能的实现都会让我兴奋好一段时间。

同时，我还要感谢提供项目需求的启辰资产管理有限公司，如果没有你们的积极配合和支持，我是无法单人解决那些苦难和疑惑，最终能够让本文顺利完成，项目成功上线。

至此论文付梓之际，我的心情无法保持平静，从开始进行需求分析到论文的顺利完成，项目的顺利上线，有许多前辈给予我很多大帮助，在这里请您接受我诚挚的谢意！

最后，再次对那些在项目完成过程中，帮助我的前辈们和朋友们表示衷心的感谢

独创性声明

本人郑重声明，所呈交的设计（论文）是我本人在指导教师的指导下进行的研究工作并取得的研究成果。尽我所知，除了中文特别加以标注和致谢的地方外，设计（论文）中没有抄袭、剽窃其他人已经发表或撰写的研究成果，也不存在为获得江西理工大学或其他教育机构的学位或证书所使用过的材料。为本设计（论文）的完成给予过的帮助、做出过的贡献均已在设计（论文）中做了明确的说明并表示了谢意。

签名： 日期：

关于设计（论文）使用授权的说明

本人完全了解江西理工大学有关保留、使用毕业设计（论文）的规定，即：学校有权保留宋文设计（论文）的原件，允许设计（论文）被查阅和借阅；学校可以公布设计（论文）的全部或部分内容，可以影印、缩印或其他复制手段保存设计（论文）。即使是保密的设计（论文）解密后也应遵守此项规定。

签名： 日期：