

北京交通大学海滨学院

毕业设计（论文）

海滨学院贫困帮扶借款交易平台的设计与实现

姓 名：胡恩会

学 号：15111034

专 业：软件工程

学院（系）：计算机与信息技术学院

指导教师：付婷婷 讲师

2019 年 5 月

摘 要

随着网络的迅猛发展和互联网金融的蓬勃发展，市面上的借款平台也是层出不穷，为了海滨学院经济困难学生的借款方便和使用安全，特此开发一套系统来为海滨学院学生提供一个可供借款交易的平台。

系统主要从以下几个方面提供服务。首先就是用户的注册和登录；再有就是邮箱绑定，手机绑定，实名认证等相关的信息认证及信息完善；还有最主要的平台提供借款，投资和还款等功能。为了维护系统的相关信息，还会由管理员管理系统中使用的相关信息。

海滨学院贫困帮扶借款交易平台的设计与实现，使用了 MVC 的开发模式，使用了 SSM 框架作为开发框架，使用 Java 编程语言进行开发，页面展示使用了 HTML 语言以及 FreeMarker 模板引擎技术。在 Windows 操作系统下使用 MySQL 作为数据库进行数据存储，使用 Tomcat 作为后台服务器来设计实现。

关键词：网络借款平台；实名认证；Java；SSM；FreeMarker

ABSTRACT

With the rapid development of the Internet and the rapid development of Internet finance, the loan platforms in the market are also emerging one after another. In order to facilitate the borrowing and security of students from the economic difficulties of the Waterfront College, a system is developed to provide a loanable transaction for the students of the Waterfront College. Platform.

The system mainly provides services from the following aspects. The first is the user's registration and login; then there are mailbox binding, mobile phone binding, real-name authentication and other related information authentication and information improvement; and the most important platform provides loans, investment and repayment. In order to maintain information about the system, the administrator also manages the relevant information used in the system.

The design and implementation of the poverty-stricken help loan trading platform of the Waterfront College uses the MVC development model, uses the SSM framework as the development framework, and uses the Java programming language for development. The page display uses the HTML language and the FreeMarker template engine technology. Use MySQL as the database for data storage under Windows operating system, and use Tomcat as the background server to design and implement.

KEYWORDS: Online loan platform; Real name certification; Java; SSM; FreeMarker

目 录

摘 要	i
ABSTRACT.....	ii
目 录	iii
1 绪论	1
1.1 项目背景及意义	1
1.2 国内外研究现状	1
1.3 主要研究内容	2
1.4 论文组织结构	2
2 系统分析	3
2.1 可行性分析	3
2.1.1 技术可行性	3
2.1.2 经济可行性	3
2.1.3 操作可行性	3
2.2 需求概述	3
2.3 功能性需求分析	4
2.3.1 登录/注册模块	4
2.3.2 邮箱绑定模块	5
2.3.3 手机绑定模块	5
2.3.4 实名认证模块	5
2.3.5 借款模块	5
2.3.6 投资模块	6
2.3.7 还款模块	6
2.3.8 后台信息维护模块	7
2.4 系统用例分析	7
2.5 非功能性需求分析	8
3 系统设计	9

3.1	系统总体设计	9
3.1.1	整体结构设计	9
3.1.2	功能性结构设计	10
3.1.3	关键技术介绍	10
3.2	系统详细设计	11
3.2.1	登录/注册模块设计	11
3.2.2	邮箱绑定模块设计	13
3.2.3	手机绑定模块设计	14
3.2.4	实名认证模块设计	16
3.2.5	借款模块设计	17
3.2.6	投资模块设计	19
3.2.7	还款模块设计	20
3.2.8	后台信息维护模块设计	22
3.3	数据库设计	22
3.3.1	概念结构设计	22
3.3.2	逻辑结构设计	23
4	系统实现	29
4.1	登录/注册模块	30
4.2	邮箱绑定模块	32
4.3	手机绑定模块	33
4.4	实名认证模块	35
4.5	借款模块	37
4.6	投资模块	39
4.7	还款模块	40
4.8	后台信息维护模块	41
5	系统测试	43
5.1	测试原则	43
5.2	功能测试	43
5.2.1	登录/注册测试报告	43

5.2.2	邮箱绑定测试报告	44
5.2.3	手机绑定测试报告	45
5.2.4	实名认证测试报告	45
5.2.5	借款测试报告	46
5.2.6	投资测试报告	46
5.2.7	还款测试报告	47
5.2.8	后台测试报告	47
6	总结和展望	48
6.1	总结	48
6.2	展望	49
	参考文献	50
	致 谢	51

1 绪论

海滨学院贫困帮扶借款交易平台的选题背景及意义，网络借款平台在国内外的研究现状以及海滨学院贫困帮扶借款交易平台的主要研究内容。

1.1 项目背景及意义

随着人民群众的生活日益改善，供孩子上大学也不再是遥不可及的梦想。对于贫困家庭的学生，学校联合银行提供助学贷款解决贫困生的学费问题，但大学生活中的生活费用对普通家庭来说仍是一笔不小的负担，很多学生省吃俭用，仍不可避免会出现资金周转困难的情况。随着网络贷款的盛行，校园贷也进入大众的视野，但它的出现不光带来了福利，也带来了威胁。考虑到学生之间的家庭经济情况不同，可以从学生之间的借款交易来解决这种资金周转困难的问题。随着互联网+时代的来临，将实际问题和互联网联系起来，可以更便捷的为大众提供服务，所以一个海滨学院贫困帮扶借款交易平台的出现，也就成为了必然。

海滨学院贫困帮扶借款交易平台，将给遇到困难的同学和愿意帮助他人的同学提供一个借款交易的平台，让遇到困难的同学可以度过难关，也让帮助他人的同学获得更好的理财。

1.2 国内外研究现状

作为一种新颖的理财模式，p2p 网络借款平台在英美等发达国家发展比较完善，被身处网络时代的大众广为接受^[1]。美国最大的网络借款平台 Prosper 于 2006 年 2 月上线运营，欧洲最大的网络借款平台 Zopa 于 2005 年 3 月在英国开始运营。这两个平台提供 p2p 的金融服务，实现用户之间资金借入或借出，整个过程无需银行的介入^[2]。p2p 网络借款理念于 2007 年传入国内，首次在上海登陆。

p2p 借款平台在国外一直比较稳定，但在国内的十多年发展中，却经历了很多波折。目前我国 p2p 网络借款平台主要有三种经营模式。第一种，拍拍贷（线上），审核方式为信用认证和信用审核；不提供担保；利率为可变利率且较高；贷款方向为线上小额。第二种，宜信（线下），审核方式为实地审核，提供担保；利率为固定利率且为低利率；贷款方向为线下额度不定。第三种，安心贷（线上线下结

合)，审核方式为实地审核、抵押担保；利率为固定利率且较低；贷款方向主要为北京地区。三种模式的区别是平台对交易的参与程度^[3]。

1.3 主要研究内容

研究海滨学院学生如何进行实名认证能够方便有效的证明他的学生身份，并且确保操作为本人操作，在确保身份认证无误之后，还要研究如何让学生之间能够进行借款，需要学生之间能够方便看到其他人的借款信息，还要能够通过简单的操作，对借款进行投资，以达到成功交易的目的。在借款成功后，需要准确无误的计算还款金额。在系统中，需要对用户的每一个状态进行区分，还要对借款交易的状态进行区分，从而保证系统安全可靠提供服务。

1.4 论文组织结构

海滨学院贫困帮扶借款交易平台论文一共有 6 章，每章的主要内容如下。

第 1 章论文的项目背景和研究现状，以及项目主要研究内容。

第 2 章海滨学院贫困帮扶借款交易平台的可行性分析及系统的需求分析。

第 3 章海滨学院贫困帮扶借款交易平台开发的总体设计和详细设计以及数据库相关设计。

第 4 章海滨学院贫困帮扶借款交易平台实现的相关过程以及实现效果。

第 5 章海滨学院贫困帮扶借款交易平台实现之后对其功能的基本测试。

第 6 章海滨学院贫困帮扶借款交易平台的总结和展望。

2 系统分析

通过对海滨学院贫困帮扶借款交易平台的可行性分析确定系统可行，并根据用户需求确定系统的功能，做出以下分析。

2.1 可行性分析

开发海滨学院贫困帮扶借款交易平台是为了给海滨学院学生提供一个投资借款平台。可行性分析从技术、经济、操作方面来进行分析，以此判断出系统开发是否可行，及系统是否能够投入使用。

2.1.1 技术可行性

使用 Java 语言开发 Web 项目，是一个很成熟的技术，有很多开源框架可以帮助开发者快速简洁的完成业务功能，Tomcat 服务器和 MySQL 数据库，操作简单，技术成熟。开发所需的技术在网上有很多技术分享可供参考，综上所述，本系统的开发是可行的。

2.1.2 经济可行性

在系统开发过程中使用的技术和框架都是免费的，只有在调用阿里云接口时候会产生少许费用，所以在经济方面分析，此系统的开发也是可行的。

2.1.3 操作可行性

系统的面向用户为海滨学院的学生，针对海滨学院学生，采取了调查问卷调研，从调研结果分析，所有的同学都会在他人在遇到困难时伸出援助之手。所以从操作方面，本系统的开发也是可行的。

2.2 需求概述

系统的主要作用是为海滨学院学生提供借款交易的平台，则系统主要功能应该包含借款、投资、还款等功能，但从用户安全方面考虑，此系统的最终用户为海滨学院学生，要想限定用户群体，必须要在注册用户的时候，对注册人信息进

行筛选，只允许符合条件的人群成为此系统的用户。考虑到可能有信息泄露导致的用户注册，在注册时期对用户的筛选只作为第一次筛选，还必须在之后的操作中，确认注册用户是本人操作，所以需要实名认证功能，用其他手段保证用户为本系统限定服务的用户。实名认证服务需要一些前置条件，包括姓名，身份证号，手机号，所以要包括手机号码的绑定。为了降低风险，对填写的身份信息，最好是有对应的实物证件照片，所以用户需要上传证件照片，并由后台管理人员审核。学生信息中，院系和专业是必不可少的信息，并且两者之间包含关联关系，对这类信息，系统中不能固定死，所以系统将此信息作为数据字典，由后台管理员管理，则需要后台信息维护功能。

2.3 功能性需求分析

对各个功能模块进行详细分析，了解各功能的业务逻辑和使用场景是功能需求分析要做的事情。

2.3.1 登录/注册模块

登录功能：则用户输入用户名和密码进行校验，对登录记录，不论成功与否，都保存登录信息，校验成功则登录成功，否则为登录失败。

注册功能：

1.平台用户角度分析：

注册功能：由于此系统面向用户为海滨学院学生，对使用平台的用户要加以限制，就要在用户注册期间，将用户进行筛选，要想实现此功能，则需要全校学生的信息，由于学生在校期间都有唯一的 id，也就是学号，所以只要有学号和姓名，就可以算作拥有了全校学生的信息，此信息由学校提供，开发阶段，只需要少许信息即可。拥有了所有学生信息，在学生注册阶段就进行填写并校验，如果注册填写的姓名和学号不在其中，则判定用户不是本校学生，不能注册成功。注册完成后，需要对系统使用中的数据进行初始化，包括可借额度等信息。

2.系统管理人员角度：

系统要求安全性较高，为了管理方便，暂只设置一个管理员，且该管理员账号不能以注册的形式创建，在系统启动时期，为系统初始化一个管理员是可行的，

这样能够避免其他人对管理员账户的注册。

2.3.2 邮箱绑定模块

邮箱绑定的目的是为了让用户能够接收到系统发送的消息提示，所以要确保绑定邮箱是真实可用，并且为用户本人的，此功能需要用户填写本人邮箱，由系统为该邮箱发送一条激活邮件，并要求用户在五天内点击邮件中的链接，完成邮箱绑定功能。

2.3.3 手机绑定模块

手机绑定是实名认证的前置条件，此功能需要使用第三方网关技术，借助第三方网关，将系统生成的验证码发送到手机号上。此功能需要用户填写手机号，然后等待系统发送短信验证码，并在五分钟内将验证码输入到系统中，完成手机绑定功能。为了防止用户恶意点击手机绑定，验证码发送功能在每次发送将有一段时间的禁用期

2.3.4 实名认证模块

实名认证功能为此系统除主要业务外最重要的功能，包括注册阶段的姓名和学号的校验，其实也属于实名的一部分，但注册阶段的校验只进行了粗略的实名，并不能准确的确定用户为海滨学院学生且为本人操作，此处，使用阿里云提供的手机号实名认证功能，此功能大致描述如下：提供用户的姓名、身份证号、手机号，平台返回判定结果，结果为一致或不一致。所以说手机绑定为实名认证的前置条件。此系统在平台判定条件下，又加入了人工审核，用户申请实名认证，需要包含系统中已有的姓名、手机号，还需要用户填写身份证号，性别，并将身份证和学生信息证件的照片上传至系统中，由后台管理人员对信息的一致性进行判定。

2.3.5 借款模块

此功能为系统的核心功能，借款功能可以分为几个部分。

1.借款前置流程，也就是用户发起借款前需要满足的条件，包括基本信息的填

写、手机号码的绑定、实名认证通过审核。还需要满足当前没有正在招标的借款项目。

2.填写借款信息并将借款信息发布。借款信息需要包括借款的金额，愿意支付的利息，借款后还款的分期月数，借款的招标天数，还有借款的标题和此次借款的描述（借款的原因和目的）。

3.借款信息需要所有人看到，所以将借款列表展示到首页，但未登录人员不可查看借款的详情，对借款信息的详情页面，需要展示借款人此次的借款信息和投资可以获得的收益，以此引起投资人的兴趣，还需要展示借款人的基本资料和联系方式，再此还展示借款人的学号，可以在线下交易阶段查看对方的证件，保证借款的安全。在借款详情页底部，展示投资记录。在交易完成之前，借款人都可以对借款进行撤销。

2.3.6 投资模块

用户除了的平台发布借款信息，还可以对其他人的借款信息进行投资，投资功能是依赖于借款功能的，所以投资功能的首页就是借款列表，但此处对借款列表进行分类查询，投资人可以根据分类查看系统的交易情况，并且可以有选择的对借款进行投资，投资功能需要进入借款详情，此处借款详情页面和借款功能处一样，只在登陆人角色不同处进行区别，借款人看到自己的借款，其他人才可以对此借款进行投资，在借款详情页展示了借款人的信息和联系方式，投资人在投资前，可以电话联系借款人，对借款人借款情况进行了解，在此期间还可以大致了解借款人的性格情况，在综合所有情况后进行选择是否投资，在一定程度上保证了借款的安全性。

投资成功后，投资人在系统的投标明细中，可以查看到自己的所有投资。

2.3.7 还款模块

还款功能的前置条件是借款与投资匹配成功，并且交易阶段没有发生取消情况，在交易成功后，系统会根据借款人的借款信息，对借款金额、利息、分期数等数据进行计算，自动生成还款计划，并展示给借款人，此信息需要包括对应的借款名称，还款的金额，分别显示本金和利息，该还款的还款期数，和还款的

最终截至日期。

借款人点击还款后，需要与投资人进行线下交易，并通知投资人进行线上确认。如果超过约定时间还没有还款，则判断用户逾期还款。

投资人在借款人进行还款操作后，可以在收款明细中看到此次收款信息，在确认收款后，线上对此笔收款进行确认。

2.3.8 后台信息维护模块

后台管理员需要对系统中使用到的信息进行维护，维护信息包括学院学生信息，学院院系信息，学院专业信息。学生信息为学校提供的全部学生姓名和学号。

2.4 系统用例分析

系统用例图如图 2-1 所示。系统包含用户和管理员两个角色，从两个角度分别分析得出系统用例图。

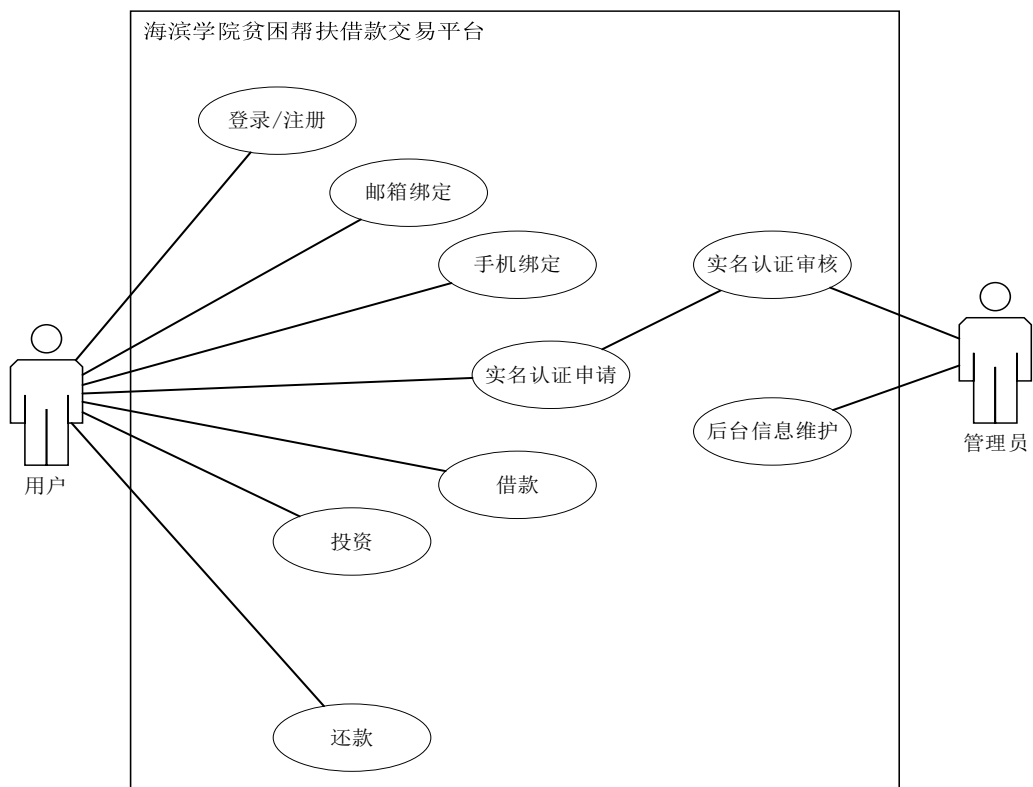


图 2-1 系统用例图

对系统用例进行分析，如表 2-1 所示。

表 2-1 系统用例分析

用例	使用语境	范围	级别	主执行者	前置条件
注册	用户没有账号	用户	用户目标	用户	用户注册
登录	用户正确输入 用户名密码	用户	用户目标	用户	用户登录
邮箱绑定	要绑定邮箱	用户	用户目标	用户	已登录
手机绑定	要绑定手机	用户	用户目标	用户	已登陆
实名认证	要实名认证	用户	用户目标	用户	已登录，已绑定手机
借款	需要借款	用户	用户目标	用户	已绑定手机，已完善 信息，已实名认证
投资	想要投资	用户	用户目标	用户	已登录
还款	借款需要还款	用户	用户目标	用户	已登录，已成功借款
后台信息 维护	信息需要维护	用户	用户目标	管理员	已登录

2.5 非功能性需求分析

系统的非功能性需求从页面美观方面考虑。

在网站开发中，界面的美观很重要，界面美观的程度直接影响着用户的第一体验效果。没有一个美观的界面会影响到网站的商业效果。网站的外观要和开发理念以及整体的文字等元素完美搭配。

3 系统设计

系统设计分为整体设计、详细设计和数据库相关的设计，整体设计描述了所使用的核心技术和系统整体技术架构以及系统的功能模块设计；详细设计对功能模块进行描述，数据库设计用实体关系和数据库表来描述。

3.1 系统总体设计

本节介绍系统整体结构设计，功能结构设计和设计中使用的关键技术，并对关键技术进行相关的描述。

3.1.1 整体结构设计

根据使用技术和系统中的功能以及使用的外部提供的服务，可以得到系统的整体结构设计图，主要体现系统所使用的技术和系统所包含的主要功能，如图 3-1 所示。

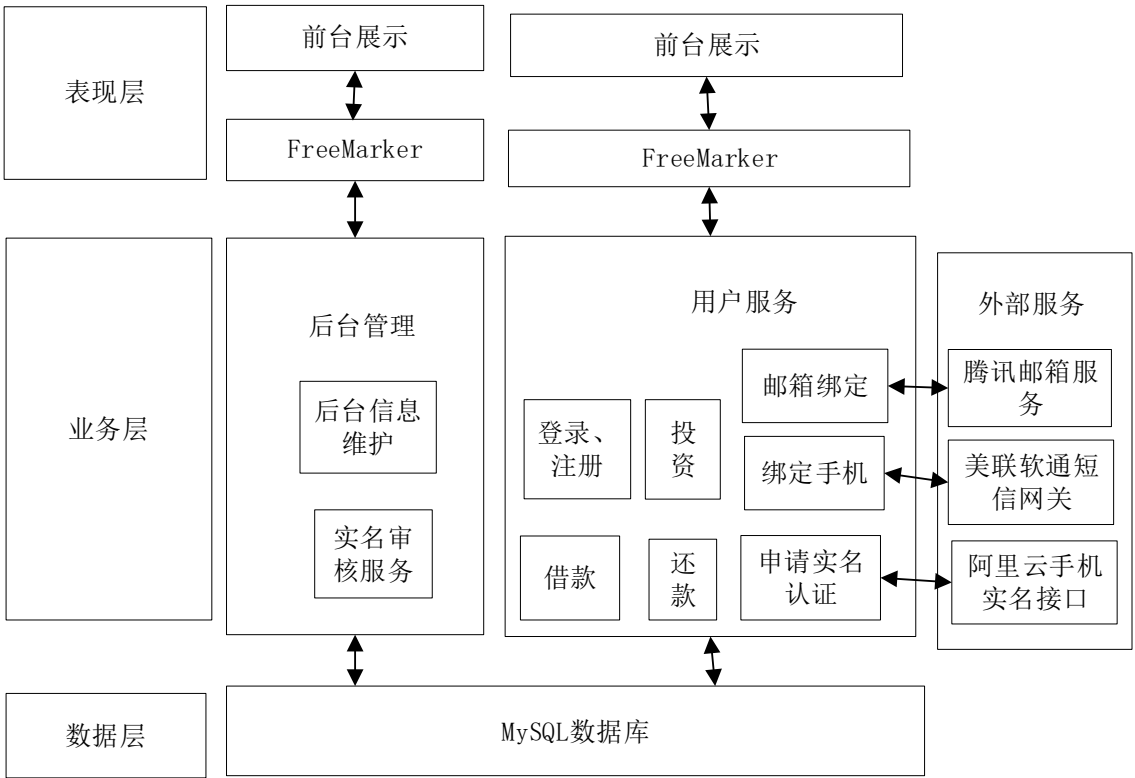


图 3-1 整体结构设计图

3.1.2 功能性结构设计

海滨学院贫困帮扶借款交易平台的功能结构设计图如图 3-2 所示。

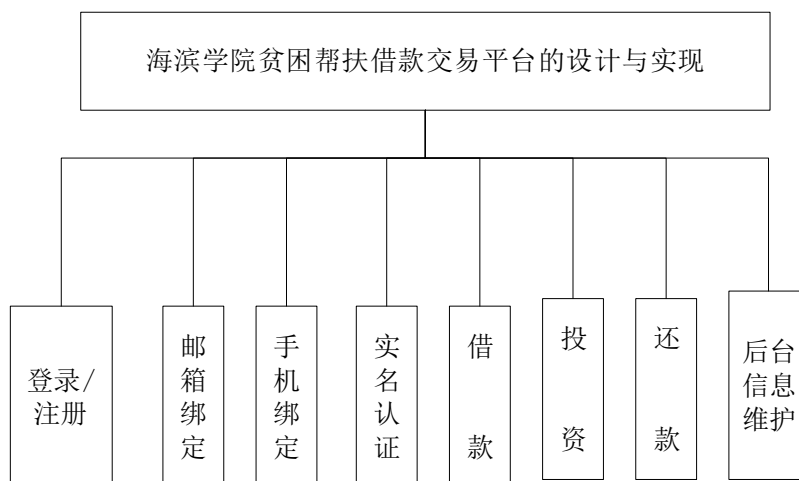


图 3-2 功能结构设计图

3.1.3 关键技术介绍

此系统使用 Java 编程语言开发，框架选择 SSM(Spring+SpringMVC+MyBatis)，使用 FreeMarker 模板引擎生成视图页面，下面详细介绍这些技术。

Java 是一门面向对象编程语言，Java 语言具有功能强大和简单易用两个特征。Java 语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程^[4]。

SSM 框架，是目前比较主流的 Java EE 企业级框架，适用于搭建各种大型的企业级应用系统^[5]。Spring 是为了解决企业应用开发的复杂性而创建的。Spring MVC 原生支持的 Spring 特性，让开发变得非常简单规范。MyBatis 是一个基于 Java 的持久层框架。MyBatis 消除了几乎所有的 JDBC 代码和参数的手工设置以及结果集的检索。MyBatis 使用简单的 XML 或注解用于配置和原始映射，将接口和 Java 对象映射成数据库中的记录。可以这么理解，MyBatis 是一个用来帮你管理数据增删改查的框架^[6]。

FreeMarker 是一个采用 Java 开发的模版引擎，是一个基于模版生成文本的通用工具。虽然 FreeMarker 具有一些编程的能力，但通常由 Java 程序准备要显示的数据，由 FreeMarker 生成页面，并通过模板显示准备的数据^[7]。

3.2 系统详细设计

本节详细描述功能模块设计，并画出相应的流程图或活动图，并画出类图和时序图。

3.2.1 登录/注册模块设计

注册功能需要用户设置用户名和密码，用来登录系统，对用户名的要求是不得与其他人的重复，所以需要用异步技术，在用户输入完成后进行提示，用户还需要输入本人姓名和学号，与系统中已存在的学生信息进行对比，只允许海滨学院学生注册。登录功能需要用户输入已注册的用户名和密码，完成校验即可登录进入系统。流程图如图 3-3 所示。

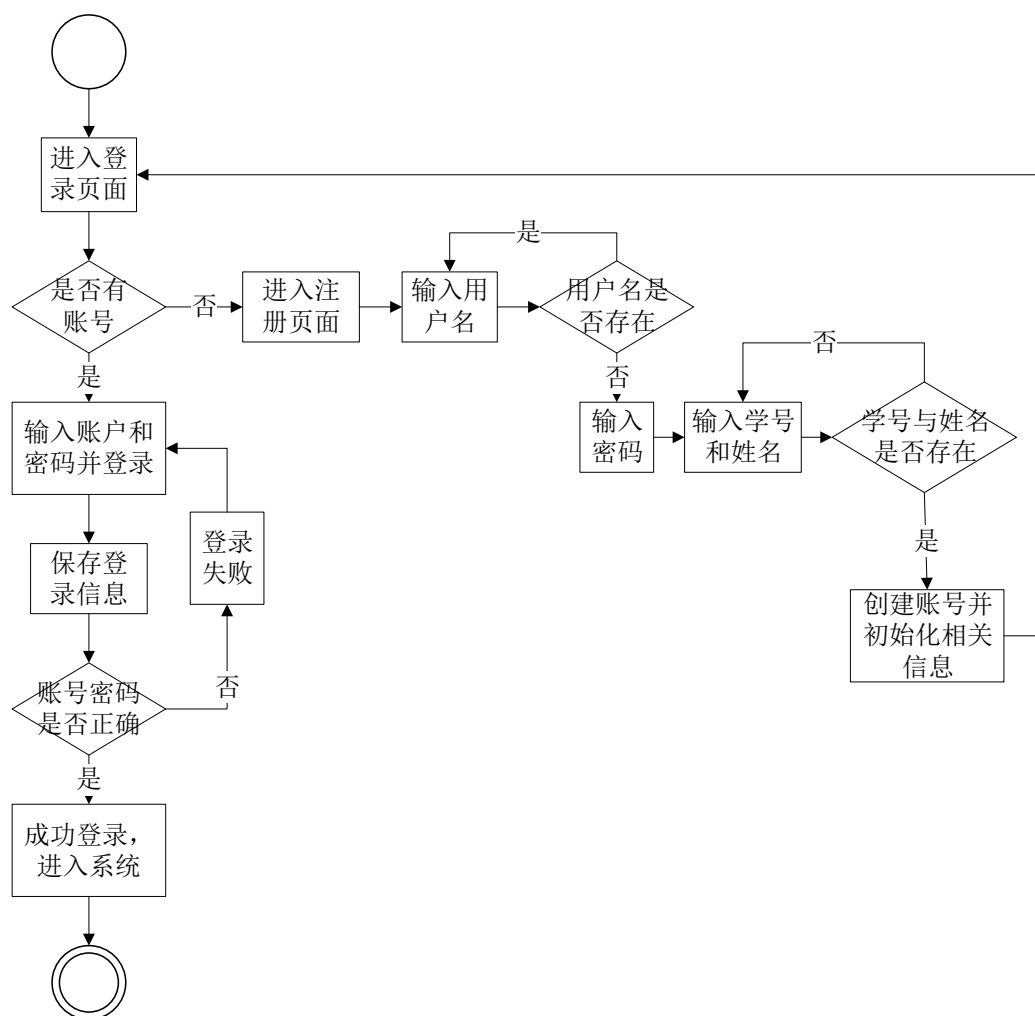


图 3-3 登录注册流程图

注册登录部分的类图如图 3-4 所示。

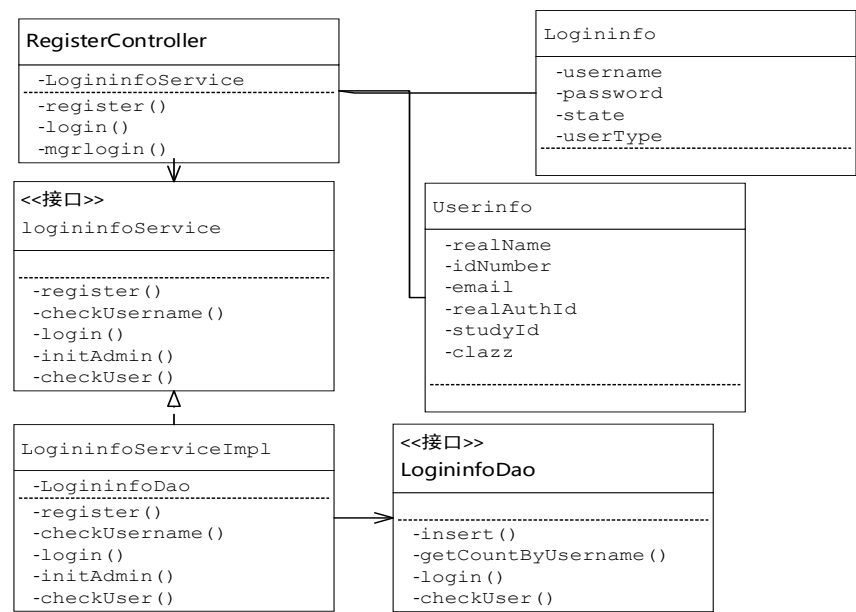


图 3-4 注册登录类图

用户注册的时序图如图 3-5 所示。

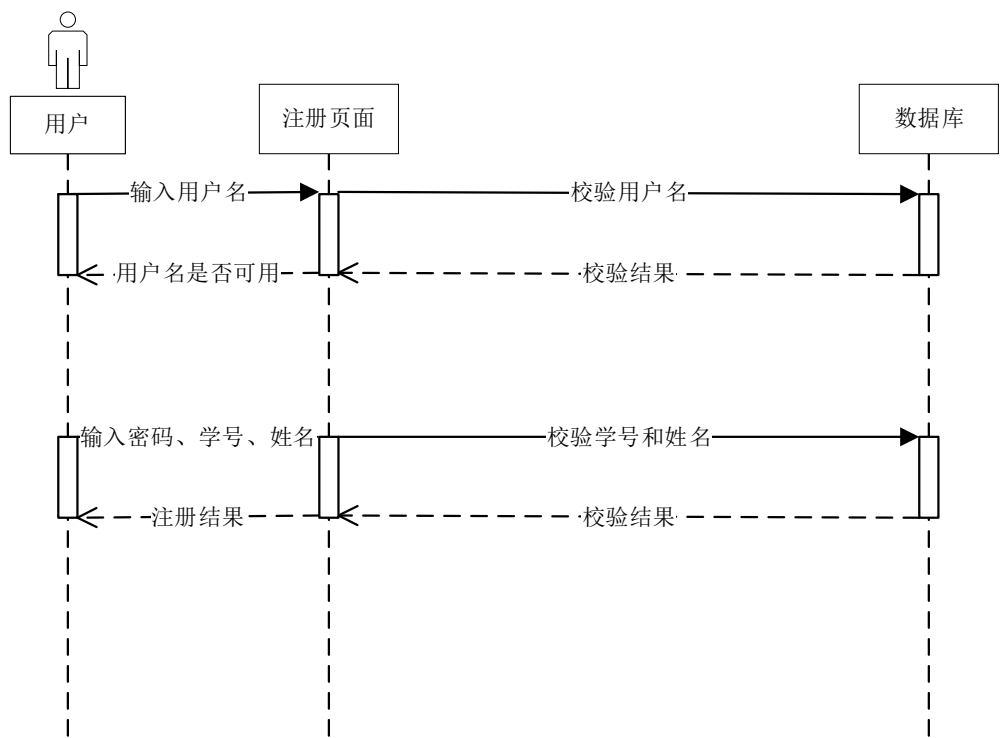


图 3-5 注册时序图

3.2.2 邮箱绑定模块设计

邮箱绑定功能需要用户输入自己的邮箱，点击保存，系统将会给填写的邮箱发送验证邮件，用户登录自己的邮箱，点击该邮件，会修改对应账号的绑定邮箱状态，完成邮箱绑定，活动图如图 3-6 所示。

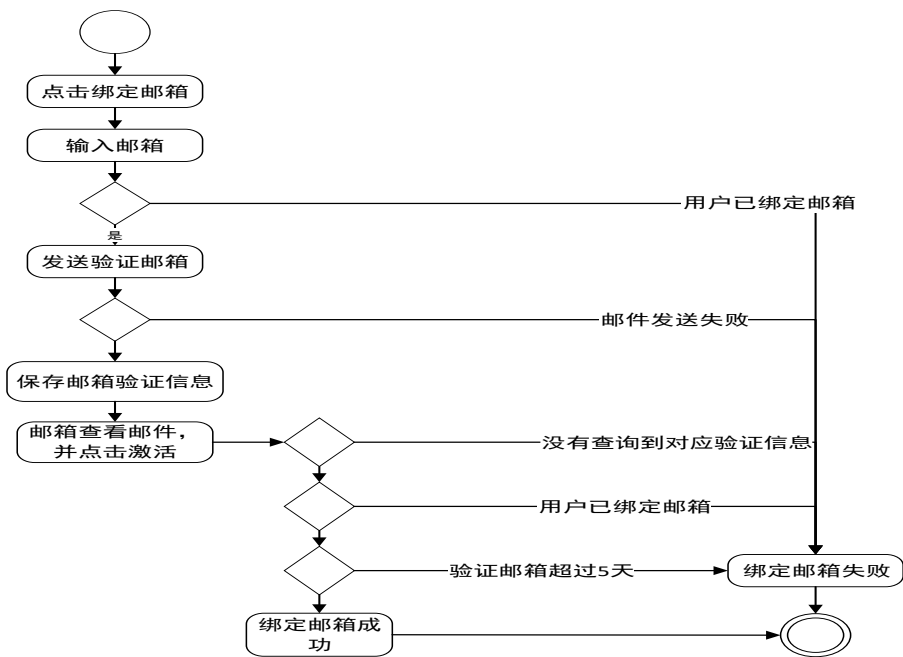


图 3-6 邮箱绑定活动图

绑定邮箱的类图如图 3-7 所示。

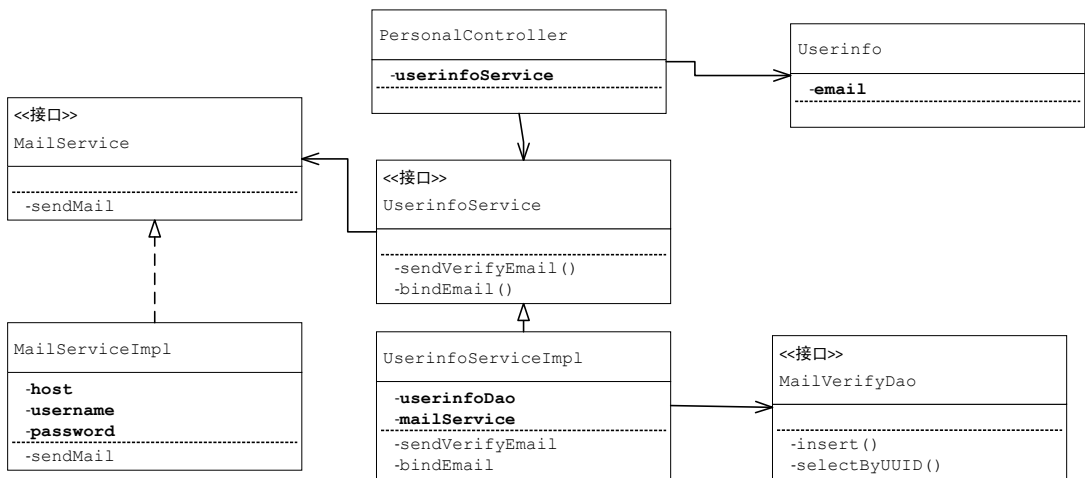


图 3-7 邮箱绑定类图

邮箱绑定的时序图如图 3-8 所示。

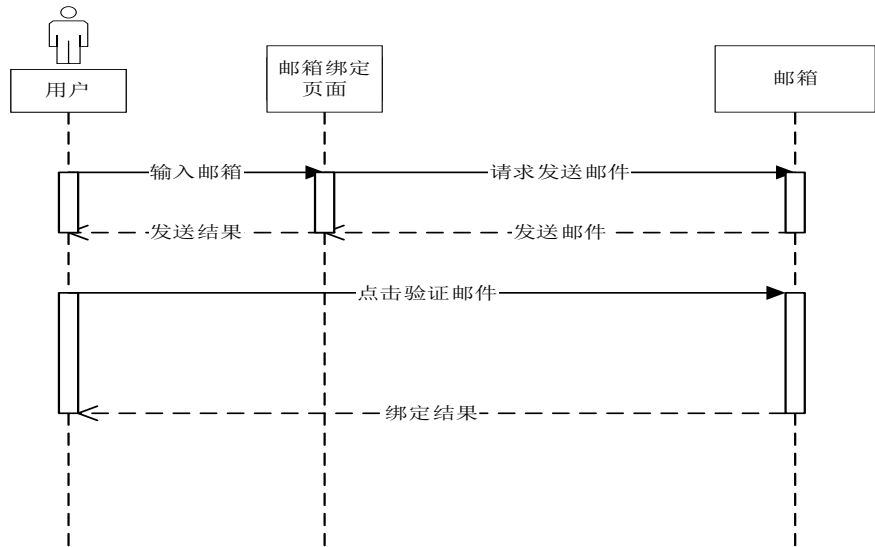


图 3-8 邮箱绑定时序图

3.2.3 手机绑定模块设计

用户点击手机绑定，弹出对话框，输入手机号点击发送验证码，系统将会给填写的手机号发送验证码，为了防止恶意操作，发送验证码功能每 90 秒可点击一次，用户需要在 5 分钟内填写手机接收到的验证码，验证码正确，则绑定手机成功。活动图如图 3-9 所示。

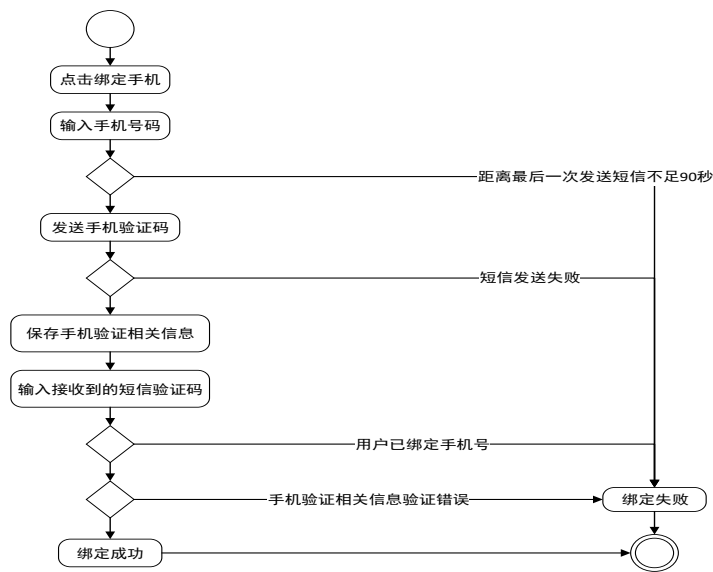


图 3-9 手机绑定活动图

手机绑定的类图如图 3-10 所示。

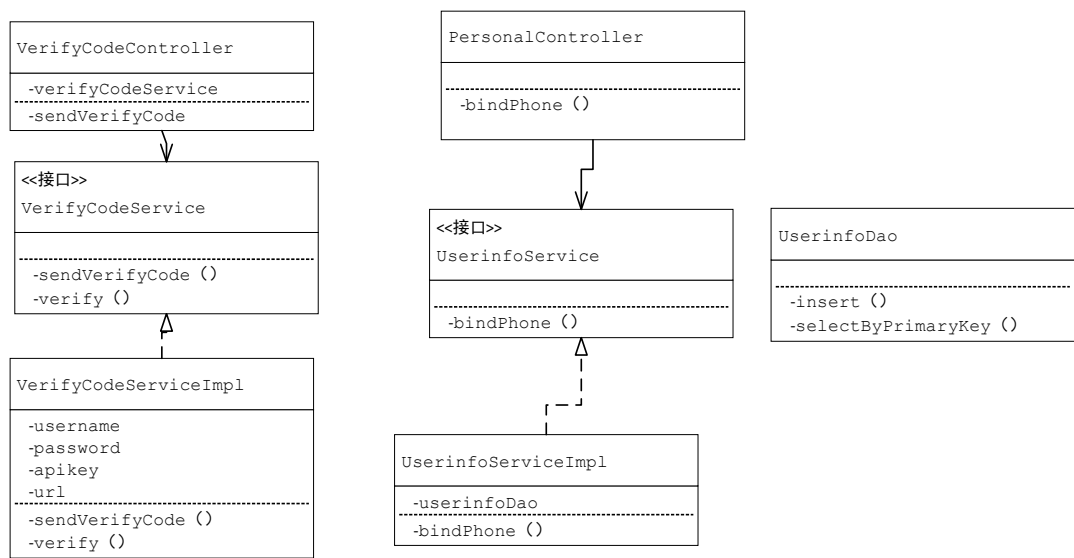


图 3-10 手机绑定类图

手机绑定的时序图如图 3-11 所示。

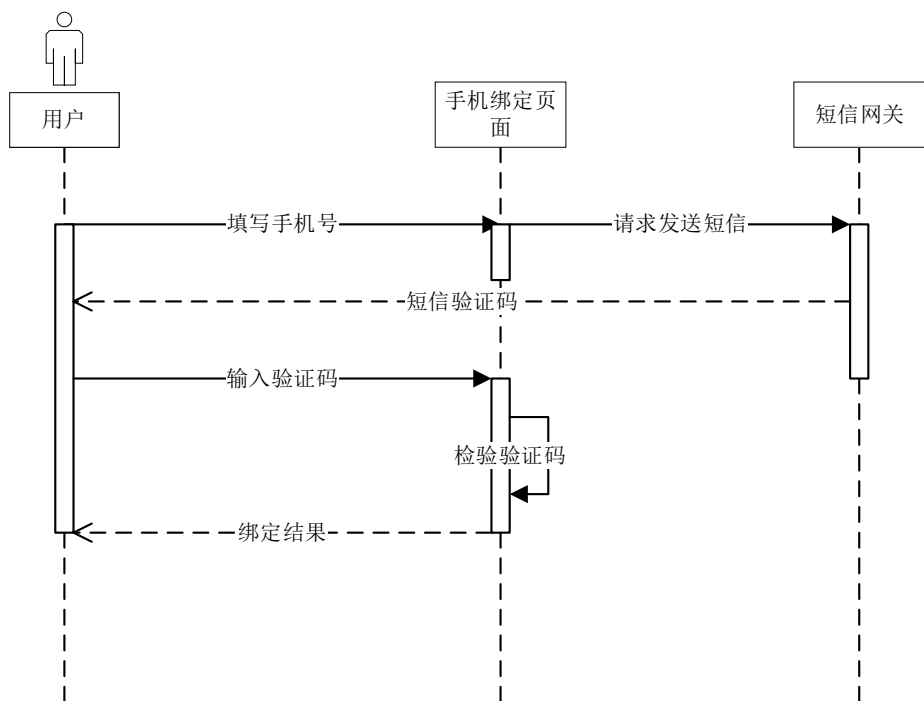


图 3-11 手机绑定时序图

3.2.4 实名认证模块设计

用户绑定手机后，可对实名认证进行申请，需要用户填写相关信息，并上传对应图片，系统首先根据姓名、手机号和身份证号，使用阿里云提供的手机实名认证功能，对填写信息进行第一次校验，校验通过后，实名申请成功，此时需要系统管理员对申请进行审核，实名认证是否成功，取决于最后管理员的审核。活动图如图 3-12 所示。

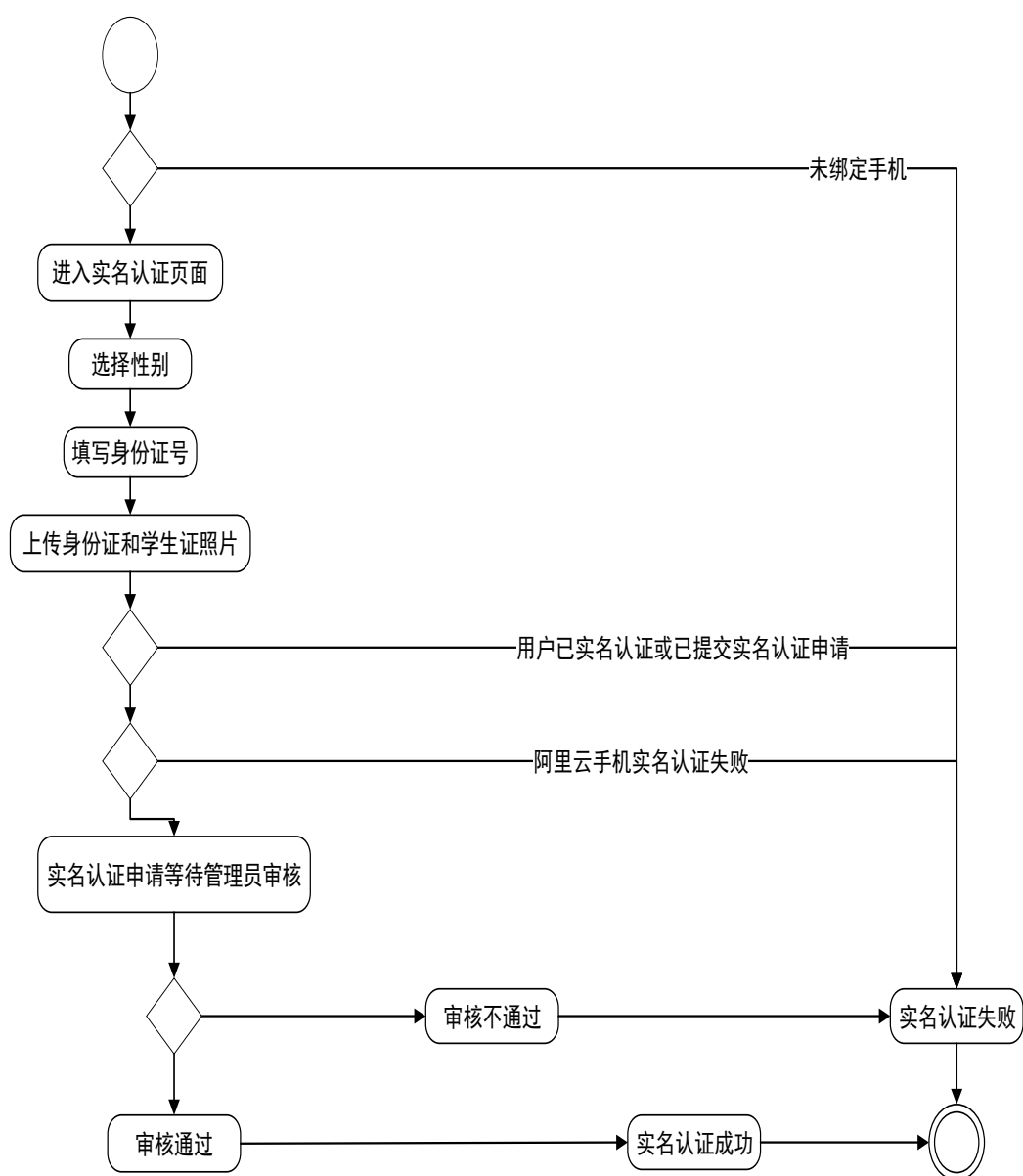


图 3-12 实名认证活动图

实名认证的类图如图 3-13 所示。

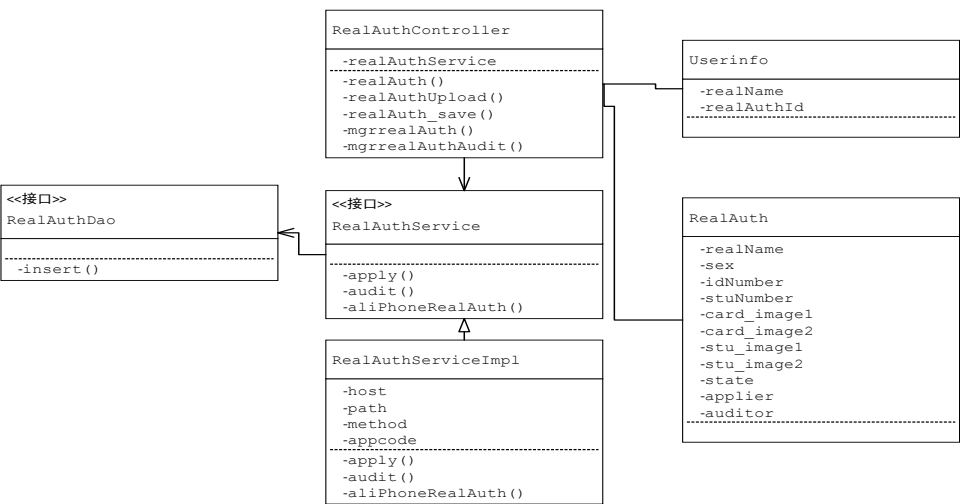


图 3-13 实名认证类图

实名认证的时序图如图 3-14 所示。

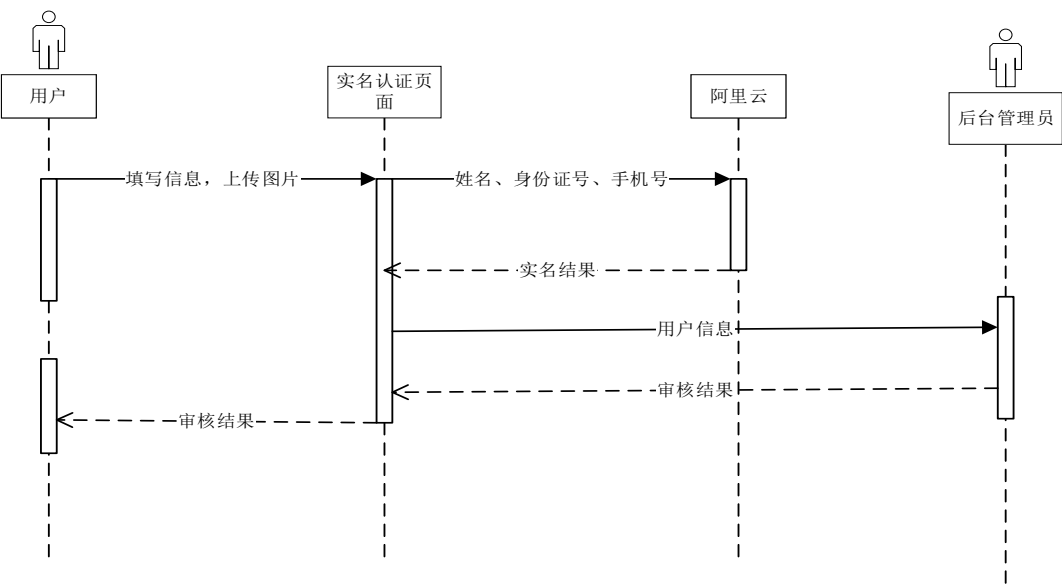


图 3-14 实名认证时序图

3.2.5 借款模块设计

对已完成手机绑定、实名认证和信息完善的用户，提供借款服务，用户可以点击我要借款按钮进入，此时系统会对用户进行拦截，完成以上前置要求才可以

进行借款，进入借款页面，需要填写借款相关信息，填写信息无误，则借款信息会发布到系统中，可以让其他用户查看，并且对此借款进行投资。活动图如图 3-15 所示。

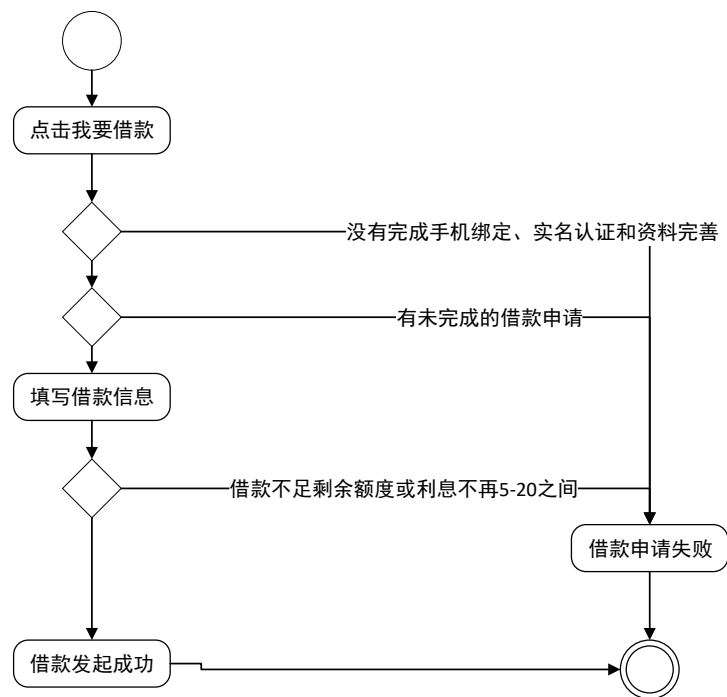


图 3-15 借款活动图

借款类图如图 3-16 所示。

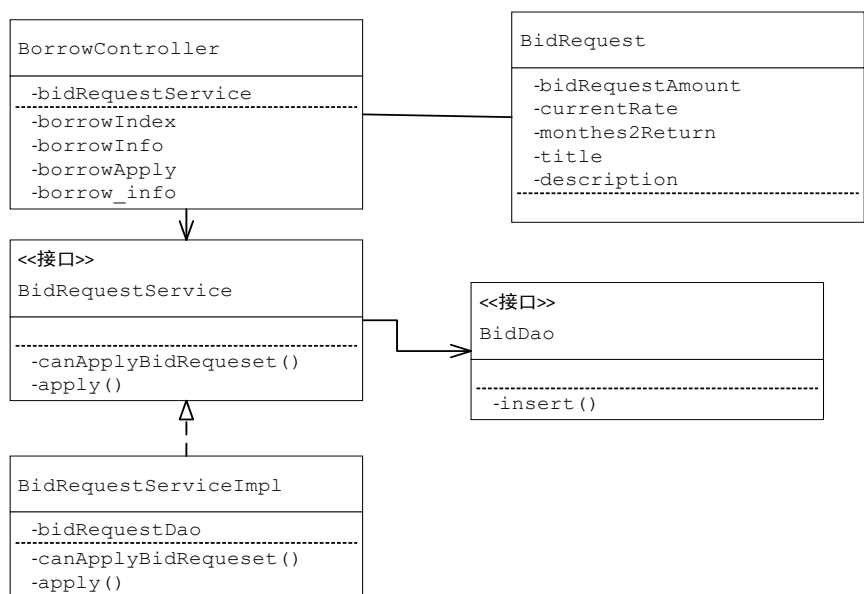


图 3-16 借款类图

借款的时序图如图 3-17 所示。

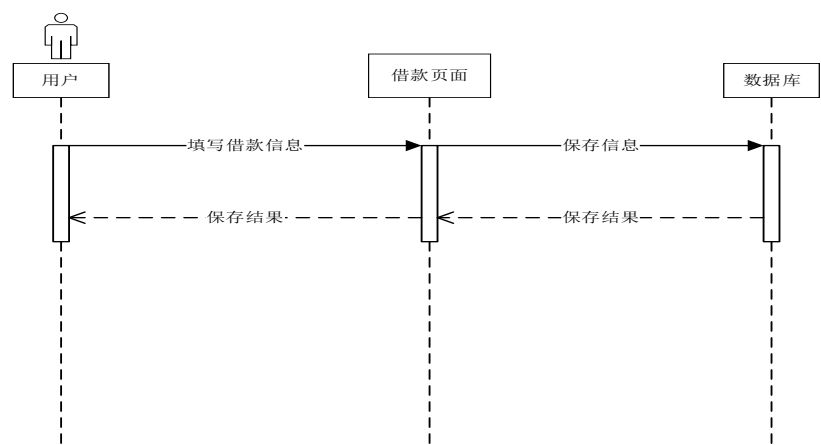


图 3-17 借款时序图

3.2.6 投资模块设计

用户点击我要投资按钮，可以看到所有正在招标的信息，可以对借款信息查看详情，详情中可以看到借款人的相关信息，和此笔借款的相关信息，有意愿对借款进行投资，可根据信息与借款人联系，也可以直接点击投资，投资后需双方交易并确认，投资活动图如图 3-18 所示。

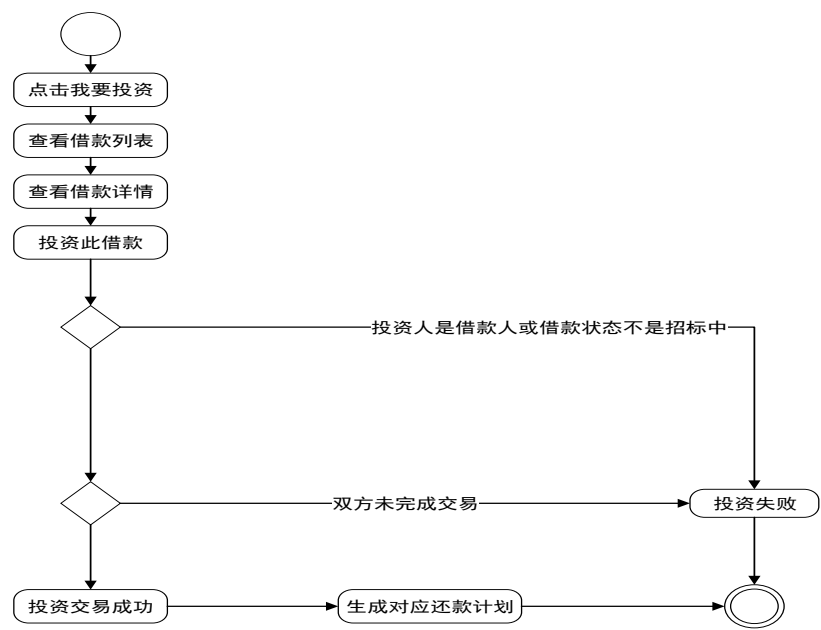


图 3-18 投资活动图

投资类图如图 3-19 所示。

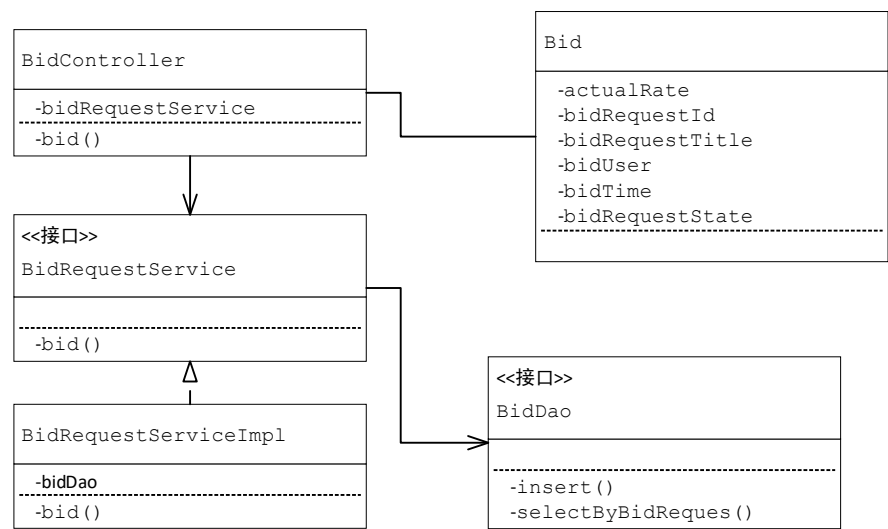


图 3-19 投资类图

投资时序图如图 3-20 所示。

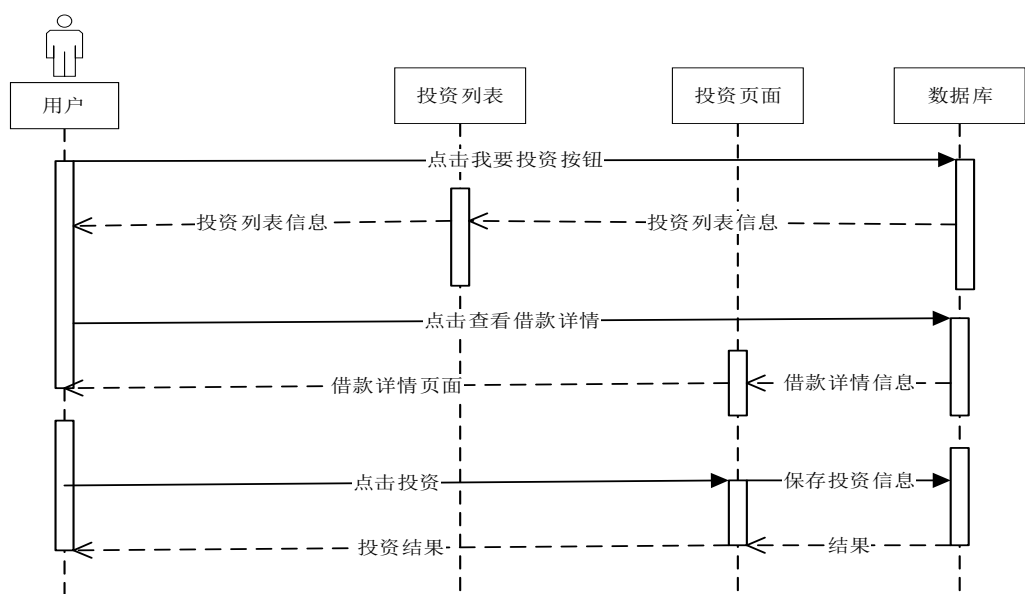


图 3-20 投资时序图

3.2.7 还款模块设计

用户申请借款并与投资人完成交易后，系统会自动根据借款时填写的信息进

行计算，按照填写的分期数和利息计算每笔还款钱数和时间，借款人可以对每一笔还款进行操作，借款人点击还款后，投资人可在收款列表查看，交易以双方确认为准，活动图如图 3-21 所示。

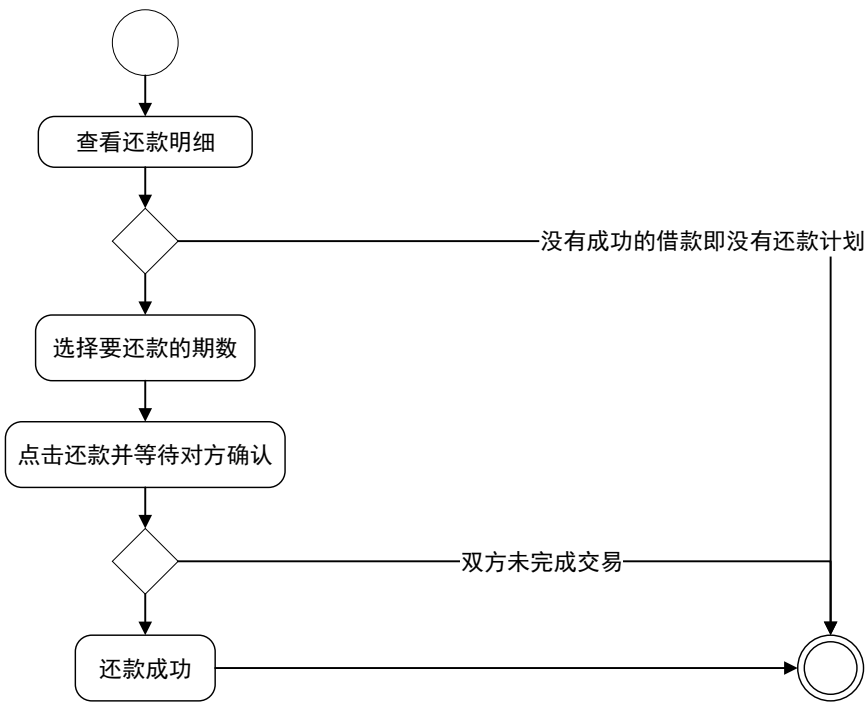


图 3-21 还款活动图

还款类图如图 3-22 所示。

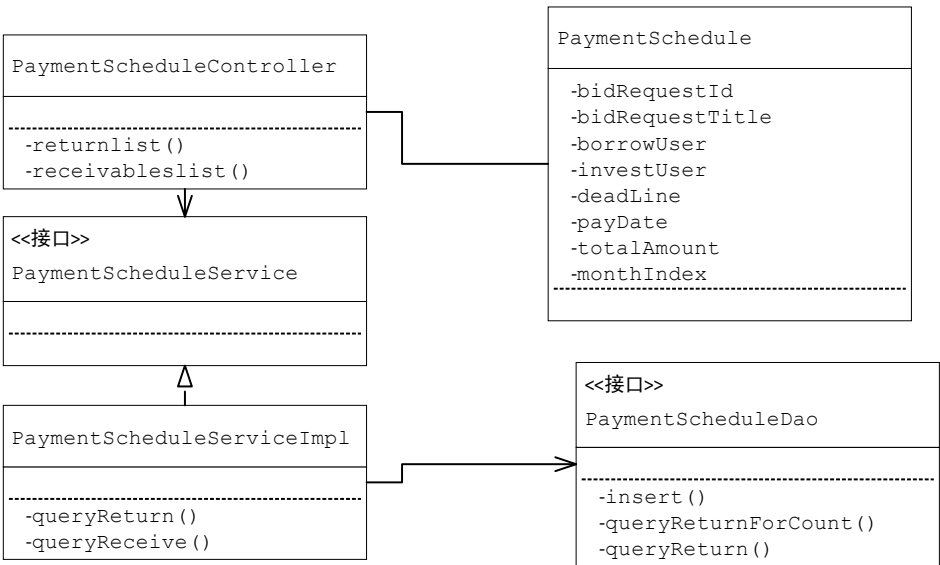


图 3-22 还款类图

还款时序图如图 3-23 所示。

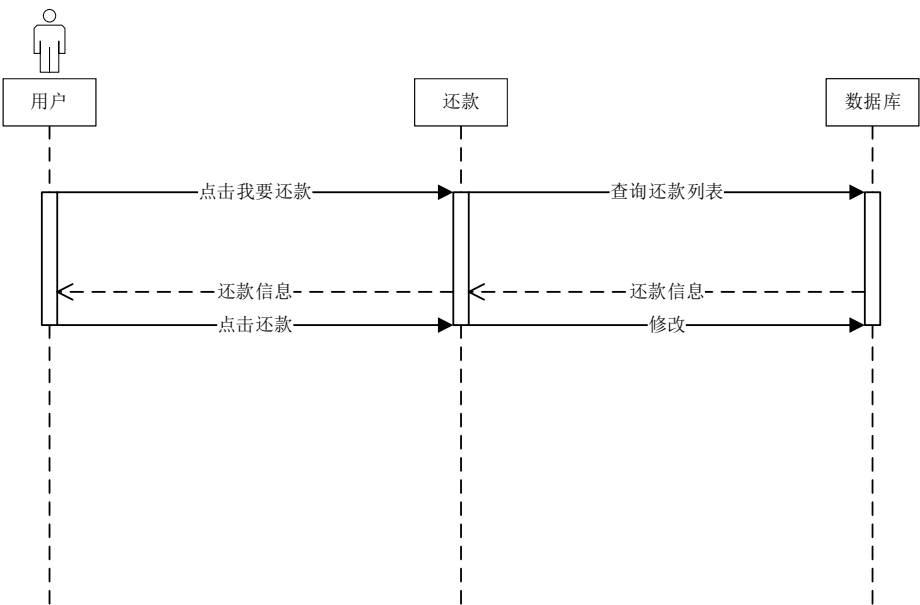


图 3-23 还款时序图

3.2.8 后台信息维护模块设计

后台信息维护包括系统管理员对学院学生、学院院系、学院专业进行查看和添加，还可以根据条件查询登录记录。

3.3 数据库设计

数据库设计从两个方面描述，首先表示概念结构设计，即 E-R 图，表示各实体间的关系，然后使用数据库表信息描述各实体所拥有的属性，能更加直观了解系统中的数据设计。

3.3.1 概念结构设计

系统的 E-R 图如图 3-24 所示，描述了用户、账号、账户、实名认证、登录记录、借款、投资、邮箱绑定、院系、专业、还款等信息之间的关系。

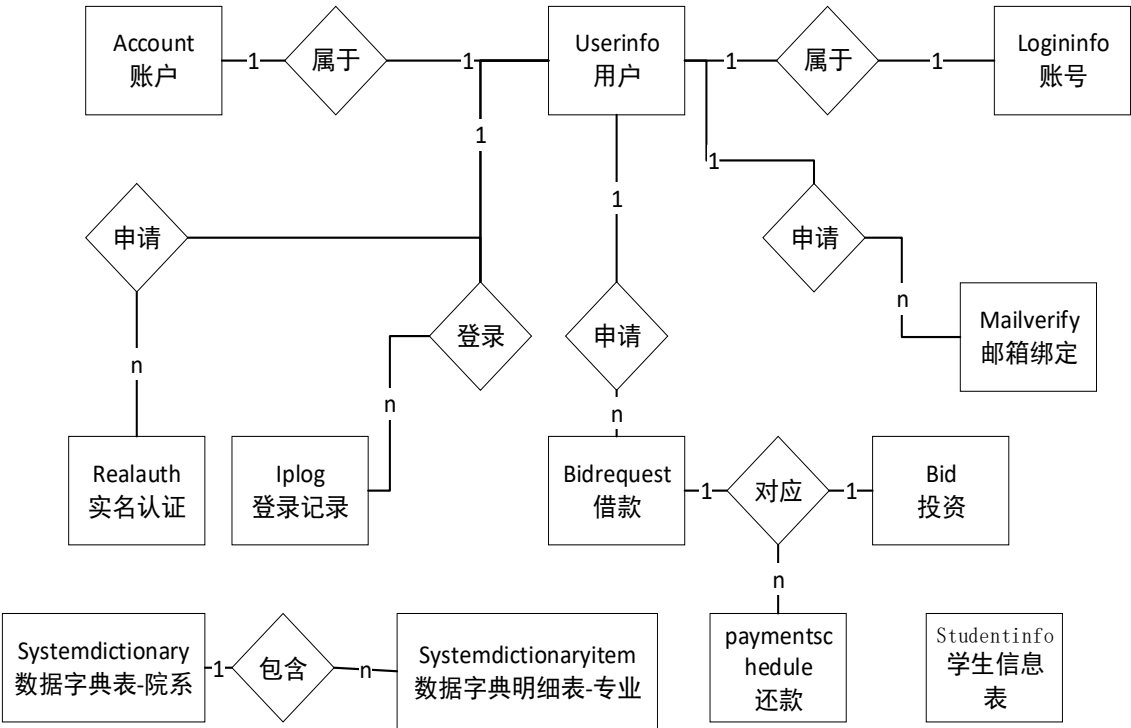


图 3-24 系统 E-R 图

3.3.2 逻辑结构设计

账户信息表（account），如表 3-1 所示。用于存储用户的账户信息，所含字段及其意义分别为：id（主键）与用户的 id 一致，borrowLimit 存储用户的最高额度，version 存储版本号（乐观锁），unReceiveInterest 存储待收利息，unReceivePrincipal 存储待收本金，unReturnAmount 存储待还金额，remainBorrowLimit 存储剩余额度。

表 3-1 账户信息表（account 表）

名称	字段名称	数据类型	主键	注释
用户 id	id	bigint(20)	是	与 userinfo id 一致
最高额度	borrowLimit	decimal(18,4)	否	最高额度
版本号	version	int(11)	否	版本号（乐观锁）
待收利息	unReceiveInterest	decimal(18,4)	否	待收利息
待收本金	unReceivePrincipal	decimal(18,4)	否	待收本金
待还金额	unReturnAmount	decimal(18,4)	否	待还金额
剩余额度	remainBorrowLimit	decimal(18,4)	否	剩余额度

投资信息表（bid），如表 3-2 所示。用于存储用户投资信息，所含字段及其意义分别为：id（主键），actualRate 存储年利率，availableAmount 存储投资金额，

bidRequest_id 存储对应借款的 id，bidUser_id 存储投资人的 id，bidTime 存储投资时间，bidRequestTitle 存储借款标题。

表 3-2 投资信息表（bid 表）

名称	字段名称	数据类型	主键	注释
主键	id	bigint(20)	是	投资主键
年利率	actualRate	decimal(8,4)	否	年利率
投资金额	availableAmount	decimal(18,4)	否	投资金额
对应借款 id	bidRequest_id	bigint(20)	否	对应借款 id
投资人 id	bidUser_id	bigint(20)	否	投资人 id
投资时间	bidTime	datetime	否	投资时间
借款标题	bidRequestTitle	varchar(255)	否	借款标题（冗余）

借款信息表（bidrequest），如表 3-3 所示。用于存储用户借款信息，所含字段及其意义分别为：id（主键），version 存储版本号（乐观锁），bidRequestType 存储借款类型，bidRequestState 存储借款状态，bidRequestAmount 存储借款金额，currentRate 存储年利率，monthes2Return 存储还款期数，totalRewardAmount 存储总回报金额（总利息），title 存储当前借款的标题，bidState 存储当前交易的状态码，disableDate 存储招标的截止日期，createUser_id 存储的使借款人的 id，disableDays 存储招标的天数，applyTime 存储申请借款的时间。

表 3-3 借款信息表（bidrequest 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
版本号	version	int(11)	否	版本号
借款类型	bidRequestType	tinyint(4)	否	借款类型
借款状态	bidRequestState	tinyint(4)	否	借款状态
借款金额	bidRequestAmount	decimal(18,4)	否	借款金额
年化利率	currentRate	decimal(8,4)	否	年化利率
还款月数	monthes2Return	tinyint(4)	否	还款月数
总回报金额	totalRewardAmount	decimal(18,4)	否	（总利息）
标题	title	varchar(255)	否	标题
交易状态码	bidState	bigint(20)	否	交易状态码
招标截至日期	disableDate	datetime	否	招标截至日期
借款人	createUser_id	bigint(20)	否	借款人
招标天数	disableDays	tinyint(4)	否	招标天数
申请时间	applyTime	datetime	否	申请时间

登录日志表（iplog），如表 3-4 所示。用于存储用户借款信息，所含字段及其意义分别为：id（主键），ip 登录人的 ip 地址，state 存储登录状态，userName 存储用户名，loginTime 存储登录时间，userType 存储用户类型。

表 3-4 登录日志表（iplog 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
登录 ip	ip	varchar(50)	否	登录 ip
登录状态	state	tinyint(4)	否	1 成功 2 失败
用户名	userName	varchar(50)	否	用户名
登录时间	loginTime	datetime	否	登录时间
用户类型	userType	tinyint(4)	否	0 后台 1 前台

登录用户表（logininfo），如表 3-5 所示。用于存储用户借款信息，所含字段及其意义分别为：id（主键），username 存储登录用户名，password 存储登录密码，state 存储账户状态，userType 存储用户类型。

表 3-5 登录用户表（logininfo 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
用户名	username	varchar(50)	否	用户名
密码	password	varchar(200)	否	密码
账户状态	state	tinyint(4)	否	0 正常 1 锁定
用户类型	userType	tinyint(4)	否	0 后台 1 前台

邮箱验证信息表（mailverify），如表 3-6 所示。

表 3-6 邮箱验证信息表（mailverify 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
用户 id	userinfo_id	bigint(20)	否	用户 id
发送时间	sendDate	datetime	否	发送时间
唯一标识	uuid	varchar(255)	否	唯一标识
邮箱号	email	varchar(255)	否	邮箱号

还款信息表（paymentschedule），如表 3-7 所示。用于存储用户还款计划的信息，所含字段及其意义分别为：id（主键），deadLine 存储本期还款的截至日期，payDate 存储还款的时间，totalAmount 存储本期还款的总金额，principal 存储本期还款的本金，interest 存储本期还款的总利息，monthIndex 存储记录本条记录的还款期数，state 存储本期还款的状态，bidRequestType 保存借款类型，returnType 保存还款类型，bidrequest_id 保存借款的 id，borrowUser_id 保存还款人的 id，bidRequestTitle 保存借款的标题，investUser_id 保存收款人的 id。

表 3-7 还款信息表（paymentschedule 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
本期还款截止期限	deadLine	datetime	否	本期还款截止期限
还款时间	payDate	datetime	否	还款时间
本期还款总金额	totalAmount	decimal(18,4)	否	本期还款总金额
本期还款本金	principal	decimal(18,4)	否	本期还款本金
本期还款总利息	interest	decimal(18,4)	否	本期还款总利息
第几期	monthIndex	tinyint(4)	否	(即第几个月)
本期还款状态	state	tinyint(4)	否	本期还款状态
借款类型	bidRequestType	tinyint(4)	否	借款类型
还款方式	returnType	tinyint(4)	否	还款方式
对应借款	bidrequest_id	bigint(20)	否	对应借款
还款人	borrowUser_id	bigint(20)	否	还款人
借款名称	bidRequestTitle	varchar(255)	否	借款名称
收款人	investUser_id	bigint(20)	否	收款人

实名认证信息表（realauth），如表 3-8 所示。用于存储用户实名认证申请的信息，所含字段及其意义分别为：id（主键），realName 存储用户的真实姓名，sex 存储用户的性别，idNumber 存储用户的身份证号，stuNumber 存储用户的学号，state 存储状态，card_image1 存储的是身份证正面图片地址，card_image2 存储的是身份证反面图片地址，stu_image1 存储的学生证件正面图片地址，stu_image2 存储学生证件反面图片地址，remark 存储审核的备注，auditTime 存储审核的时间，applyTime 存储申请的时间，auditor_id 存储审核人的 id，applier_id 存储申请人的 id。

表 3-8 实名认证信息表（realauth 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	tinyint(20)	是	主键 id
真实姓名	realName	varchar(50)	否	真实姓名
性别	sex	tinyint(4)	否	性别
证件号码	idNumber	varchar(50)	否	证件号码
学生号码	stuNumber	varchar(50)	否	学生号码
状态	state	tinyint(4)	否	状态
身份证正面	card_image1	varchar(255)	否	正面地址
身份证反面	card_image2	varchar(255)	否	反面地址
学生证正面	stu_image1	varchar(255)	否	正面地址
学生证反面	stu_image2	varchar(255)	否	反面地址
审核备注	remark	varchar(255)	否	审核备注
审核时间	auditTime	datetime	否	审核时间
申请时间	applyTime	datetime	否	申请时间
审核人	auditor_id	bigint(20)	否	审核人
申请人	applier_id	bigint(20)	否	申请人

学生官方信息表（studentinfo），如表 3-9 所示。用于存储海滨学院学生的官方信息。Stuid 存储学号，stuname 存储姓名。

表 3-9 学生官方信息表（studentinfo 表）

名称	字段名称	数据类型	主键	注释
学号	stuid	varchar(255)	是	学号
真实姓名	stuname	varchar(255)	否	真实姓名

数据字典表（systemdictionary），如表 3-10 所示。用于存储海滨学院院系信息。Id（主键），sn 存储字典分类编码，title 存储字典分类标题（院系名称）。

表 3-10 数据字典表（systemdictionary 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
字典分类编码	sn	varchar(50)	否	字典分类编码
字典分类名称	title	varchar(50)	否	字典分类名称

数据字典表明细表（systemdictionaryitem），如表 3-11 所示。用于存储海滨学院专业信息。Id（主键），parentId 存储数据字典分类的 id，title 存储数据字典明细

的标题（专业名称），sequence 存储数据字典明细的序号。

表 3-11 数据字典表明细表（systemdictionaryitem 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
所属分类 id	parentId	bigint(20)	否	所属分类 id
字典明细名称	title	varchar(50)	否	字典明细名称
字典明细序号	sequence	tinyint(4)	否	字典明细序号

用户信息表（userinfo），如表 3-12 所示。用于存储用户信息。Id（主键），version 存储版本号（乐观锁）bitState 存储用户状态码，realName 存储用户的真实姓名，idNumber 存储用户的身份证号，phoneNumber 存储用户的手机号，email 存储用户的邮箱号，department_id 存储学生所在院系的 id，profession_id 存储学生所在专业的 id，studyId 存储学号，clazz 存储学生所在班级，realAuthId 存储正在实名认证申请的 id。

表 3-12 用户信息表（userinfo 表）

名称	字段名称	数据类型	主键	注释
主键 id	id	bigint(20)	是	主键 id
版本	version	int(11)	否	版本
用户状态码	bitState	bigint(20)	否	用户状态码
真实姓名	realName	varchar(30)	否	真实姓名
身份证号	idNumber	varchar(30)	否	身份证号
手机号	phoneNumber	varchar(30)	否	手机号
邮箱	email	varchar(255)	否	邮箱
院系	department_id	bigint(20)	否	院系
专业	profession_id	bigint(20)	否	专业
学号	studyId	varchar(30)	否	学号
班级	clazz	int(11)	否	班级
实名认证 id	realAuthId	bigint(20)	否	实名认证 id

4 系统实现

本章介绍系统功能的实现，并且通过对编写代码的展示做出了系统设计的逻辑思路，通过对每个页面的展示，做出了相应得功能。



图 4-1 个人中心页面图

用户登录后会进入个人中心页面，个人中心页面展示用户的简单信息，和账户的金额信息，手机绑定、邮箱绑定、实名认证功能也可以从个人中心页设置，个人中心页面如图 4-1 所示。个人资料页面完善学生基本信息，个人资料页面如图 4-2 所示。



图 4-2 个人资料页面图

4.1 登录/注册模块

注册页面如图 4-3 所示，在头部显示海滨学院字样图片，并标注此功能为注册功能，对填写信息在输入框下面标出提示，填写完成点击同意协议并注册就可以注册平台的用户。

The screenshot shows a registration page for Beijing Jiaotong University Hai Bin College. At the top, there is a navigation bar with links for '首页' (Home), '登录' (Login), and '帮助' (Help). Below this, the page title '用户注册' (User Registration) is displayed. The main content area contains a registration form with the following fields and prompts:

- 用户名 (Username):** Input field with 'huenhui123'. Prompt: '用户名为4~16位字母, 数字, 符号或中文' (Username must be 4~16 letters, numbers, symbols, or Chinese).
- 密码 (Password):** Input field with masked characters. Prompt: '密码为4~16位字符组成, 采用数字、字母、符号安全性更高' (Password must be 4~16 characters, using numbers, letters, and symbols for higher security).
- 确认密码 (Confirm Password):** Input field with masked characters. Prompt: '请再次填写密码' (Please re-enter the password).
- 学号 (Student ID):** Input field with '15111034'.
- 真实姓名 (Real Name):** Input field with '胡恩会'.

At the bottom of the form, there is a green button labeled '同意协议并注册' (Agree to terms and register) and a link '已有账号, 马上登录' (Already have an account, login now). A link to the '《使用协议说明书》' (User Agreement) is also present.

4-3 注册页面图

注册功能核心代码如下所示，首先查看用户名是否存在，不存在才可以注册，注册需要保存用户信息，并且初始化账户信息和用户信息。

```
int count = this.logininfoDao.getCountByUsername(username);
if(count<=0){
    Logininfo li = new Logininfo();
    this.logininfoDao.insert(li);
    Account account = new Account();
    this.accountService.add(account);
    Userinfo userinfo = new Userinfo();
    this.userinfoService.add(userinfo);
}else{
    throw new RuntimeException("用户名已存在");
}
```

登录页面如图 4-4 所示，头部显示海滨学院字样图片，并标注此功能为用户登录，登录按钮右侧有注册按钮，如果没有平台账户，则点击注册按钮，去注册一个用户在登录。



图 4-4 登录页面图

登录功能核心代码如下所示，根据用户输入的用户名和密码去数据库查看，并且创建登录记录对象保存此次登录记录，如果输入的数据正确，就登录成功，并且把用户信息保存到 session 中，否则就返回错误提示。

```
Logininfo current = logininfoDao.login(username,MD5.encode(password));
Iplog iplog = new Iplog();
if(current != null){
    iplog.setState(Iplog.STATE_SUCCESS);
    UserContext.putCurrent(current);
}else{
    iplog.setState(Iplog.STATE_FAILED);
}
this.iplogDao.insert(iplog);
return current;
```

用户登录，系统会记录登录操作信息，用户可以查看登录记录，用户可以查看自己的登录记录，后台管理员可以查看所有人的登录记录，前台用户能够根据时间范围查询，还可以查询登录成功的记录和登录失败的记录，查询出来的数据分页展示，每页五条数据，如图 4-5 所示。

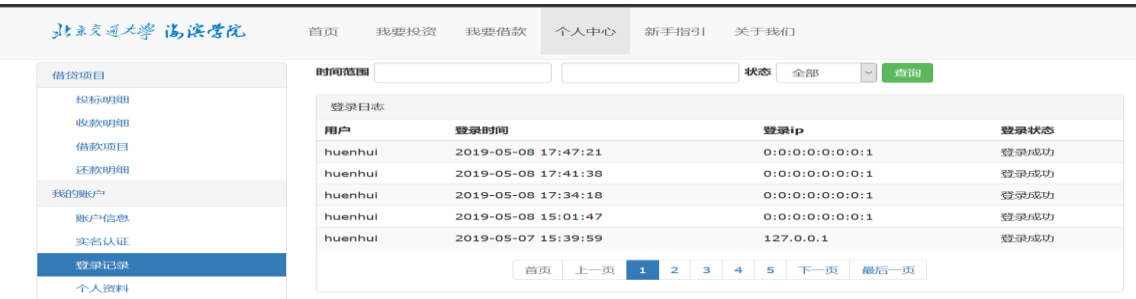


图 4-5 登录记录页面图

4.2 邮箱绑定模块

邮箱绑定在系统中的体现就是用户输入自己的邮箱，如图 4-6 所示。邮箱绑定模块可以分为两部分，一部分是发送邮件，一部分是邮箱激活。发送邮件就是用户填写邮箱并点击保存，系统会对填写的邮箱发送邮件。邮箱激活就是用户进入自己的邮箱，点击系统发送的邮件完成激活。

发送邮箱的核心代码如下。

```
Userinfo userinfo = this.getUserContext.getCurrent().getId());
if(!userinfo.getIsBindEmail()){
    String uuid = UUID.randomUUID().toString().replace("-", "");
    StringBuilder content = new StringBuilder(100)
        .append("【海滨贫困帮扶平台】点击

```

图 4-6 邮箱绑定页面图

邮箱激活的核心代码如下。

```
MailVerify mailVerify = this.mailVerifyDao.selectByUUID(uuid);
if(mailVerify != null) {
    Userinfo userinfo = this.get(mailVerify.getUserinfoId());
    if (!userinfo.getIsBindEmail()) {
        if (DateUtil.secondsBetween(mailVerify.getSendDate(), new Date())
            <= BidConst.VERIFYEMAIL_VAILDATE_DAY * 24 * 60 * 60) {
            userinfo.addState(BitStatesUtils.OP_BIND_EMAIL);
            userinfo.setEmail(mailVerify.getEmail());
            this.update(userinfo);
            return;
        }
    }
}
throw new RuntimeException("绑定邮箱失败！");
```

4.3 手机绑定模块

手机绑定模块在系统中的体现就是填写手机号，输入验证码，如图 4-7 所示。其中用户输入手机号后，系统给该手机号发送短信验证码，和用户输入验证码后，点击保存，系统验证用户填写的验证码是否正确。

发送短信验证码的核心代码如下。

```
VerifyCodeVO vc = UserContext.getCurrentVerifyCode();
if(vc==null||DateUtil.secondsBetween(new Date(),vc.getLastSendTime())>90){
    String verifyCode = UUID.randomUUID().toString().substring(0,4);
    try {
        URL url = new URL(this.url);
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        StringBuilder content = new StringBuilder(100)
            .append("username=").append(username)
            .append("&password_md5=").append(password)
            .append("&apikey=").append(apikey).append("&mobile=")
            .append(phoneNumber).append("&encode=UTF-8")
            .append("&content")
            .append("【海滨贫困帮扶平台】验证码是:");
    }
}
```

```

        .append(verifyCode)
        .append(",请在 5 分钟内使用");
conn.setRequestMethod("POST");
conn.setDoOutput(true);
conn.getOutputStream().write(content.toString().getBytes());
String response=StreamUtils.copyToString(conn.getInputStream()
        ,Charset.forName("UTF-8"));
if(response.startsWith("success:")){
    vc = new VerifyCodeVO();
    vc.setLastSendTime(new Date());
    UserContext.putVerifyCode(vc);
}else{
    throw new RuntimeException();
}
} catch (Exception e) {
    e.printStackTrace();
    throw new RuntimeException("发送短信失败! ");
}
}else{
    throw new RuntimeException("发送过于频繁");
}
}

```



The image shows a web interface for binding a mobile phone. It has a title bar '绑定手机' (Bind Mobile) with a close button. The main area contains two input fields: '手机号:' (Mobile Number) and '验证码:' (Verification Code). Below the mobile number field is a blue button labeled '发送验证码' (Send Verification Code). At the bottom right, there are two buttons: '取消' (Cancel) and '保存' (Save).

图 4-7 手机绑定页面图

验证用户输入的验证码，绑定手机号的核心代码如下。

```
Userinfo current = this.get(UserContext.getCurrent().getId());
if(!current.getIsBindPhone()){
    boolean ret = this.verifyCodeService.verify(phoneNumber,verifyCode);
    if(ret) {
        current.addState(BitStatesUtils.OP_BIND_PHONE);
        current.setPhoneNumber(phoneNumber);
        this.update(current);
    }else{
        throw new RuntimeException("绑定手机失败");
    }
}
```

4.4 实名认证模块

实名认证模块包括用户提交实名认证申请和管理员审核实名认证，实名认证申请如图 4-8 所示，实名认证审核如图 4-9 所示。

实名认证申请，用户填写信息，然后上传证件照片，点击提交，即可完成申请。申请的实名认证信息，首先通过阿里云手机实名认证服务验证信息是否一致，通过信息认证的申请将进入管理员审核阶段，如果信息认证失败，则直接判断实名认证失败。

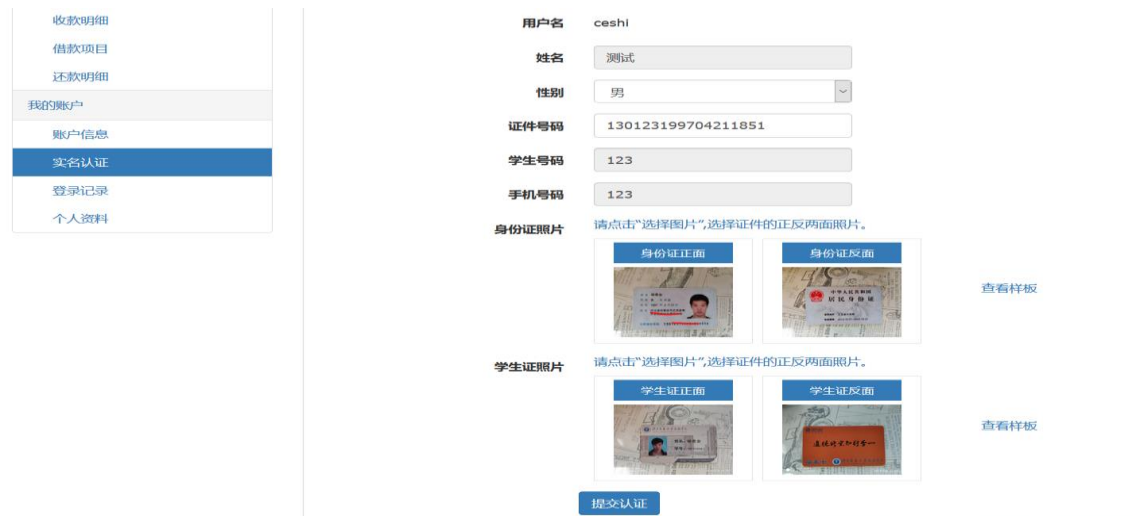


图 4-8 实名认证申请页面图

实名认证申请的核心代码如下。

```
Userinfo current = userinfoService.get(UserContext.getCurrent().getId());
if(!current.getIsRealAuth() && current.getRealAuthId() == null){
    realAuth.setApplyTime(new Date());
    realAuthDao.insert(realAuth);
    current.setRealAuthId(realAuth.getId());
    userinfoService.update(current);
}
```

使用阿里云的手机实名认证接口的核心代码如下。

```
Map<String, String> headers = new HashMap<String, String>();
headers.put("Authorization", "APPCODE " + appcode);
headers.put("Content-Type", "application/x-www-form-urlencoded;
    charset=UTF-8");
Map<String, String> querys = new HashMap<String, String>();
Map<String, String> bodys = new HashMap<String, String>();
bodys.put("idNo", idNumber);
bodys.put("name", realName);
bodys.put("phoneNo", phoneNumber);
try {
    HttpResponse response = HttpUtils.doPost(host, path, method
        , headers, querys, bodys);
    System.out.println(response.toString());
    JSONObject jsonObject = JSONObject
        .parseObject(EntityUtils.toString(response.getEntity(), "UTF-8"));
    String respCode = jsonObject.get("respCode").toString();
    if("0000".equals(respCode)){
        return true;
    }else{
        return false;
    }
} catch (Exception e) {
    e.printStackTrace();
}
return false;
```

平台管理

- 学院学生管理
- 学院院系管理
- 学院专业管理
- 登录历史

审核项目

- 实名认证审核

实名认证管理

状态全部申请时间到查询

用户名	真实姓名	性别	身份证号码	学生证号码	状态	审核人
huenhui	胡恩会	男	130123199704251512	15111034	审核通过	admin
liuhe	刘鹤	女	130123199703031512	15111035	审核通过	admin

首页上一页1下一页最后一页

图 4-9 实名认证审核页面图

4.5 借款模块

借款模块包括借款申请和借款查看，借款申请页面如图 4-10 所示，用户输入要借款的金额，借款利息，利息要求在 5 到 20 之间，设置分期还款月数，填写招标天数，和借款标题和描述，然后点击提交，将借款信息发布到平台，供其他人进行投资。借款列表页面如图 4-11 所示，展示用户发布的借款，也可以根据借款的状态进行查询。

借款信息 信用标

借款金额1000元

借款利息5%

借款期限3月

还款方式按月分期

招标天数3天

借款标题借钱买资料

借款描述本月生活费透支，现需要1000元购买课程资料，会分三期还款，希望大家帮忙

提交申请

图 4-10 借款申请页面图

借款申请的核心代码如下。

```
Account account = this.accountService.getCurrent();
```

```

if (this.canApplyBidRequest(UserContext.getCurrent().getId())) {
    BidRequest bidRequest = new BidRequest();
    bidRequest.setBidRequestAmount(br.getBidRequestAmount());
    bidRequest.setCurrentRate(br.getCurrentRate());
    bidRequest.setDescription(br.getDescription());
    bidRequest.setDisableDays(br.getDisableDays());
    bidRequest.setMonthes2Return(br.getMonthes2Return());
    bidRequest.setTitle(br.getTitle());
    bidRequest.setApplyTime(new Date());
    bidRequest.setPublishTime(new Date());
    bidRequest.setDisableDate(new Date(bidRequest.getPublishTime()
        .getTime()+bidRequest.getDisableDays()*24*60*60*1000));
    bidRequest.setBidRequestState(BidConst.BIDREQUEST_STATE_BIDDING);
    bidRequest.setCreateUser(UserContext.getCurrent());
    bidRequest.setTotalRewardAmount(CalculatetUtil.calTotalInterest(
        bidRequest.getReturnType(),
        bidRequest.getBidRequestAmount(),
        bidRequest.getCurrentRate(),
        bidRequest.getMonthes2Return()));
    this.bidRequestDao.insert(bidRequest);
    Userinfo userinfo = this.userinfoService.getCurrent();
    userinfo.addState(BitStatesUtils.OP_HAS_BIDREQUEST_PROCESS);
    this.userinfoService.update(userinfo);
}

```

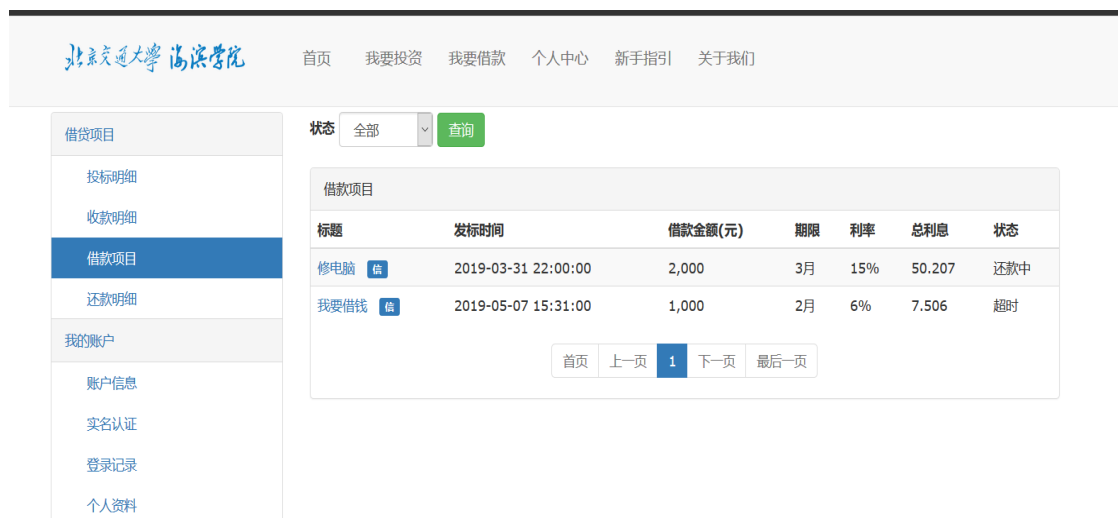


图 4-11 借款列表页面图

4.6 投资模块

投资模块首先是查看投资的列表，如图 4-12 所示，可以点击某个借款查看详情，借款详情页面如图 4-13 所示，其中包括借款人的信息，借款的信息。用户还可以在个人中心查看自己投资过的列表，如图 4-14 所示。



图 4-12 投资列表页面图

投资功能核心代码如下。

```
BidRequest br = this.get(bidRequestId);
if(满足条件){
    Bid bid = new Bid();
    bidDao.insert(bid);
    br.setBidRequestState(BidConst.BIDREQUEST_STATE_WAIT_CONFIRM);
    this.update(br);
}
```



图 4-13 借款详情页面图



图 4-14 个人投资列表页面图

4.7 还款模块

借款完成后，系统自动生成还款计划，借款人可以查看还款明细，也可以对单笔还款进行操作，还款后，投资人可以查看收款明细，进行操作上的确认，还款明细和收款明细都可以根据时间范围来查询，也都可以根据状态来查询。还款明细页面如图 4-15 所示，收款明细页面如图 4-16 所示。



图 4-15 还款明细页面图

核心代码如下。

```
Long userId = UserContext.getCurrent().getId();
qo.setUserId(userId);
int count = this.paymentScheduleDao.queryReturnForCount(qo);
```

```
if(count>0){
    List<PaymentSchedule> list = this.paymentScheduleDao.queryReturn(qo);
    return new PageResult(list,count,qo.getCurrentPage()
        ,qo.getPageSize());
}
return PageResult.empty(qo.getPageSize());
```



图 4-16 收款明细页面图

4.8 后台信息维护模块

后台信息维护主要包括学院学生管理、学院院系管理和学院学生管理。登录历史查看同前台功能一致，则此处不做介绍。学院学生管理就是对系统官方库进行维护，对学院在读学生进行操作。学院院系管理如图 4-17 所示，其他功能均相似页面，只展示其一。



图 4-17 学院院系页面图

核心代码如下。

```
int count = this.systemDictionaryDao.queryForCount(qo);
if(count>0){
    List<SystemDictionary> list = this.systemDictionaryDao.query(qo);
    return new PageResult(list,count,qo.getCurrentPage(),qo.getPageSize());
}
return PageResult.empty(qo.getPageSize());
```


5 系统测试

首先介绍测试原则，然后根据功能模块进行设计测试用例，进行功能测试，最终得到测试报告展示。

5.1 测试原则

站在用户的角度对产品进行全面测试就是软件测试的原则，还要负责跟踪和分析产品中的问题，对不足之处提出质疑和改进意见^[8]。在软件测试过程中，应注意和遵循的具体原则，可以概括为十大项：1、所有测试的标准都要基于用户需求。2、软件测试必须基于“质量第一”的思想 3、事先定义好产品的质量标准。4、软件项目一启动，软件测试也就开始，不要等程序写完，才开始进行测试。5、穷举测试是不可行的。6、第三方测试会更客观，更有效。7、软件测试计划是做好软件测试工作的前提。8、测试用例是设计出来的，不是写出来的。9、不可将测试用例置之度外，排除随意性。10、对发现错误较多的程序段，应进行更深入的测试^[9]。

5.2 功能测试

功能测试是为了检测功能是否实现预期目标，根据具体功能设计测试用例并总结出测试报告展示在本节中。测试在 Windows10 系统进行，使用 MySQL 数据库，jdk1.8，开发调试工具为 IDEA。测试策略为人工手动测试。

5.2.1 登录/注册测试报告

登录功能测试，主要测试用户名和密码检验，合法用户可以成功登录系统，错误的信息不能登录系统。登录测试报告如表 5-1 所示。

表 5-1 登录测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
系统登录	用户名和密码输入、合法性检查	huenhui, hue nhui	登录成功，跳转到主面	同预期结果

注册功能测试，主要测试填写信息的校验，包括用户名和密码长度在 4-16 之间，两次输入密码一致。学号和姓名一致且在本校数据库中存在。注册测试报告如表 5-2 所示。

表 5-2 注册测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
系统注册	用户名、密码、确认密码、学号和姓名输入、合法性检查	hu, 123, 123, 15111034, 胡恩会	用户名和密码长度不符	同预期结果
		huenhui, 12345, 23455, 15111034, 胡恩会	两次密码不一致	同预期结果
		huenhui, 123456, 123456, 1511034, 胡恩会	注册成功	同预期结果

5.2.2 邮箱绑定测试报告

邮箱绑定功能主要包括邮箱发送和邮件激活，主要测试系统能否根据用户输入的邮箱成功发送验证邮件，和用户接收到邮件点击激活，是否能成功绑定邮箱。邮箱绑定的测试报告如表 5-3 所示。

表 5-3 邮箱绑定测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
邮箱绑定		huenhui	验证邮箱发送失败	同预期结果
	邮件能否发送和邮箱能否激活	717253212@qq.com	邮件发送成功	同预期结果
		5 天内登录邮箱点击邮件	账户成功绑定邮箱	同预期结果
		5 天后登录邮箱点击邮件	邮箱激活超时，绑定邮箱失败	同预期结果

5.2.3 手机绑定测试报告

手机绑定功能主要包括发送短信功能和验证码校验功能，主要测试系统能否成功发送短信验证码，以及能否正确校验验证码的正确性，测试报告如表 5-4 所示。

表 5-4 手机绑定测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
手机绑定功能	短信能否成功发送，验证码能否正确验证	12345678	发送短信失败	同预期结果
		18894961902	成功发送短信验证码	同预期结果
		18894961902, aaaa	绑定失败	同预期结果
		18894961902, e35a	绑定成功	同预期结果

5.2.4 实名认证测试报告

实名认证功能主要包括实名认证申请、阿里云接口的检测和管理员的审核，主要测试系统能否根据用户输入的手机号、姓名、身份证号成功验证实名信息，以及管理员审核后数据的正确性，测试报告如表 5-5 所示。

表 5-5 实名认证测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
实名认证功能	阿里云接口检测是否正确，审核后数据是否正确	Ceshi, 18894961902,	不能通过阿里云	同预期测试结果
		13012319970421151X	检测	
		胡恩会, 18894961902,	实名认证进入审	同预期测试结果
		13012319970421151X	核阶段	
		点击审核通过	实名认证成功	同预期测试结果
		点击审核不通过	实名认证失败	同预期测试结果

5.2.5 借款测试报告

借款功能主要包括借款申请前置要求和借款，主要测试系统能否根据用户使用情况判断能否借款，发起借款后数据是否正确，测试报告如表 5-6。

表 5-6 借款测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
借款	借款前置条件和借款数据是否正确	没有完善信息或没有绑定手机或没有实名认证；有正在申请的借款没有交易完成	不能申请借款	同预期测试结果
		1000, 2, 借钱, 没钱了	利息要求 5-20 之间	同预期测试结果
		5000, 10, 借钱, 没钱了	借款金额超过可用额度	同预期测试结果
		1000, 10, 借钱, 没钱了	借款申请成功	同预期测试结果

5.2.6 投资测试报告

投资功能主要包括投资人点击投资，和交易阶段双方点击确定或取消是否成功，测试报告如表 5-7 所示。

表 5-7 投资测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
投资	投资按钮是否生效，交易阶段取消是否正确	借款人点击取消	借款取消	系统报错
		借款人点击取消	借款取消	同预期测试结果
		投资人点击投资	投资成功，进入交易阶段	同预期测试结果
		交易阶段点击取消	投资取消	同预期测试结果

5.2.7 还款测试报告

还款功能需要测试借款后是否成功生成还款计划，点击还款后能否成功进入交易阶段，并完成还款，测试报告如表 5-8 所示。

表 5-8 还款测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
还款	还款计划能否生成，	借款交易完成	生成还款计划	同预期结果
	点击还款是否生效	点击投资	进入交易，投资按钮生效	同预期结果

5.2.8 后台测试报告

后台包括学生信息，院系信息，专业信息的管理，主要包括添加和修改，测试报告如表 5-9 所示。

表 5-9 后台测试报告

测试项目	测试目的	测试输入	预期测试结果	实际测试结果
后台功能	后台信息添加修改是否正确	15111022，测试	学生信息添加成功	同预期结果
		计算机科学与技术， Computer Science and Technology	院系信息添加成功	同预期结果
		软件工程，1	专业信息添加成功	同预期结果

6 总结和展望

最后来总结一下此次海滨学院贫困帮扶借款交易平台的设计与实现过程中的经历和收获，以及设计中遇到的各种问题，并简要描述系统的不足之处和对改进方案做一个展望。

6.1 总结

在老师和同学的帮助下完成了这次的毕业设计——海滨学院贫困帮扶借款交易平台的设计与实现。依据任务书上的进度安排，按时准确的完成了毕业设计。在毕业设计前期准备时，复习了各门相关课本，有《语言程序设计》、《Java 开发从入门到实战》、《Spring 实战》等，在学习了这些专业知识，并且自己借阅了大量相关书籍后，对专业知识及企业开发的认识有了很大的提升，于是开始搜寻有关网络借款相关的资料，并且看了一些有关网贷平台项目的案例，在网络课程中，学习了其他网贷平台的开发思路，结合海滨学院实情，将网络贷款平台与海滨学院相结合，有了此次设计的思路。在设计初期，遇到了一个很大的问题，由于支付功能需要备案，并且需要公司的认证，所以线上交易是不可行的，在听取了老师和同学的意见后，决定采用线下交易的方式，但为了保障平台使用的绿色纯净，以防海滨学院学生以外的人群冒充学生使用平台，所以要求对实名认证方面的问题得到解决。首先的思路是由学生上传学生资料和证件信息由管理人员审核，但此方案遭到了老师的否定，因为管理人员的工作量是非常大的，并且在不了解所有学生真实信息的情况下，很难由上传资料判定信息是否真实，所以此次设计遇到了瓶颈，在搜寻了其他网贷平台的思路后，仍没有想到解决方案，于是请教老师，老师给出方案，要预先收集好学生信息，在学生认证时，首先与数据库信息进行匹配，匹配成功则认定为真实学生，老师的思路给了我一定的启发，但我又找到了一些漏洞，由于学生信息是可能会泄露的，所以单纯凭借信息填写是不能完全判定的，所以设定多次验证，首先与数据库进行比较，然后将填写信息保存入库，接下来用户绑定手机，最后填写证件信息和证件照片，通过三次操作收集到的信息，借助阿里云的手机实名认证接口判定学生信息是否真实存在，然后管理员审核证件信息和证件照片是否一致，从而最终认证用户身份，给予用户使用

平台借款的资格。最大的问题解决后，后面的设计就比较顺畅了。

下面总结一下在本设计开发过程中主要功能的实现方法，该设计整体使用 MVC 设计模式实现。其中包含了很多界面，每个界面都是使用 Bootstrap 和 JQuery 实现。其中在 JQuery 里遇到了许多难点，就是关于 JQuery 的异步请求和表单操作，经过一系列的查阅资料最终我找到了方法，使用 JQuery 可以减少开发成本，利用整理好的类库，可以更便捷的开发出漂亮的界面。在后台开发中，我尝试了 MyBatis 的注解开发，也学习了 Spring 的注解开发，在了解了注解开发后，更体会到了 XML 的臃肿。所以通过对本设计的实践，我明白了软件开发最重要的前提就是做好需求工作仔细阅读开发文档，并且也要提高自主学习的能力和解决问题的能力。

6.2 展望

对于海滨学院贫困帮扶借款交易平台的设计，由于时间和技术的不足，其实还有很多不够完善的地方，后期还可以对系统进行优化和改进。目前系统中使用线下交易，虽然在安全方面有了一定保障，但在交易的整个操作上却加大了难度，由于需要线下确认，所以在交易阶段会有很多状态需要注意，如果将系统改进为线上交易，则可以一个人借款，多个人投资，对于一笔金额较大的借款，多个人同时投资，可以加大交易的成功率，还可以增加交易的保障，容错性大大提升，对于逾期或坏账问题分摊给了多个人，系统可用性将会提升。增加并发量的两种方案，即水平扩展与垂直扩展。垂直扩展又分为，增强单机硬件性能，例如：增加 CPU 核数如 32 核，升级更好的网卡如万兆，升级更好的硬盘如 SSD，扩充硬盘容量如 2T，扩充系统内存如 128G；以及提升单机架构性能，例如：使用 Cache 来减少 IO 次数，使用异步来增加单服务吞吐量，使用无锁数据结构来减少响应时间。对于水平扩展，可以将系统分模块构建，将系统后台和前台交易分开部署，在较大请求压力下，可以集群前台服务器，增加并发访问量。已达到用户同时使用而不会导致服务器崩溃的现象^[10]。

参考文献

- [1] P2P 网络借款[J]. 金融与经济, 2013, (12): 87.
- [2] 谢广营, 徐二明. P2P 网贷行业现状、问题及管控研究[J]. 武汉金融, 2016, (11).
- [3] 张梦晶. 我国 P2P 网络借贷平台的运营模式研究[D]. 安徽财经大学, 2015.
- [4] 宋才玉. Java 语言在计算机软件开发中的应用[J]. 科技资讯, 2019, 17(05): 23-24.
- [5] Laura S. Kaiser, FACHE, President and CEO, SSM Health.[J]. Journal of healthcare management / American College of Healthcare Executives, 2019, 64(1).
- [6] 曾艳丽, 李诺. 针对 SSM 框架 Web 系统的相关思考[J]. 信息与电脑 (理论版), 2019, (03): 116-117.
- [7] 谢满, 魏小锐. 基于 FreeMarker 自定义访问控制的设计与实现[J]. 东莞理工学院学报, 2015, 22(05): 32-35.
- [8] 周秋丹. 软件测试在 IBM 域管理系统中的应用[D]. 大连理工大学, 2006.
- [9] G. Mersch,B. De Decker,C. Dobbels,L. Tremmerie,V. Kostanjevecki. Evaluation of the INNO-LiPA CFTRiage test principle[J]. Clinica Chimica Acta, 2019, 493.
- [10] 李科伟. 互联网中高并发技术架构实践[J]. 数字通信世界, 2019, (03): 65-66.

致 谢

本毕业设计（论文）的工作是在 付婷婷老师的悉心指导下完成的，付婷婷 老师严谨的治学态度和科学的工作方法给了我极大的帮助和影响。在此衷心感谢四年来付婷婷 老师对我的关心和指导。

付婷婷 老师悉心指导我们完成了毕业设计（论文），在学习上和生活上都给予了我很大的关心和帮助，在此向付婷婷 老师表示衷心的感谢。

付婷婷 老师对于我的毕业设计（论文）都提出了许多的宝贵意见，在此表示衷心的感谢。

在撰写毕业设计（论文）期间，霍欣泽 、刘鹤 等同学对我毕业设计（论文）中的借款平台 研究工作给予了热情帮助，在此向他们表达我的感激之情。

另外也感谢家人胡金梁 ，他们的理解和支持使我能够在学校专心完成我的学业。

