****大学

毕业设计（论文）

题　　目：转账系统安全研究与应用实现

学　　院：*********

专　　业：****************

学生姓名：*** 班级/学号 ***************

指导老师/督导老师：*******

起止时间：******

摘　要

随着我国的信息技术和科技的迅猛发展，网络渗透到每个人的生活当中，与此同时，人们的支付方式也不一样了，在网上交易带来了方便、快捷这些好处之余，不容忽视的是仍然存在信息的泄露和诸多安全问题，困扰着交易的安全。

本课题针对信息的安全存储问题，对信息加密技术的原理与流程进行了研究与设计。本系统是在 Windows10 环境下以及 Eclipse，MySQL 等开发工具的基础上，采用 MVC 模式，使用 Java 编程语言设计和实现的模拟转账系统。首先对比各种加密算法，从三种最具代表性的算法 AES，RSA，MD5 中选择了 AES，MD5 加以结合实现，其次设计 MySQL 数据表与相应的功能模块，最后进行功能模块界面的设计与链接。研究结果表明，本课题研究了转账服务的类型、过程及安全方法，和对于相关加密算法解密算法的研究与应用实现，使得用户能够在 web 网页上进行线上转账，实现模拟转账业务，又保障信息的保密，解决了用户信息存储泄露等不安全因素的问题，对线上转账的安全方面有十分重要的意义。

本论文详细而具体地介绍了本系统的设计与实现过程。

关键词：信息泄露；模拟转账系统；信息加密；AES；MD5；MVC 模式

Abstract

With the rapid development of information technology and technology in China, the Internet has penetrated into everyone's life. At the same time, people's payment methods are also different. The convenience of online transactions brings convenience and speed. These benefits cannot be ignored. What remains is that there are still information leaks and many security issues that plague transaction security.

This subject researches and designs the principle and process of information encryption technology in view of the problem of the safe storage of information. This system is a simulated transfer system designed and implemented in the Windows 10 environment and on the basis of Eclipse, MySQL and other development tools, using the MVC model and using the Java programming language. First, compare various encryption algorithms, choose AES, MD5 from the three most representative algorithms AES, RSA, MD5 to combine and realize, secondly design MySQL data table and corresponding function module, and finally design and interface the function module link. The research results show that this topic studies the type, process and security method of the transfer service, as well as the research and application implementation of the relevant encryption algorithm decryption algorithm, which enables users to make online transfers on the web page, realize the simulated transfer business, and guarantee The confidentiality of information solves the problem of insecure factors such as leakage of user information storage, and is of great significance to the security of online transfers.

This paper introduces the design and implementation process of this system in detail and specifically.

Keywords: information leakage; analog transfer system; information encryption; AES; MD5; MVC mode

目　　录

第一章　概述

随着我国的信息技术的快速发展，人们的生活急剧网络化，我国网民数量也大幅增长，人均上网时长显著提高，网上转账业务相应越来越频繁，对转账安全的需求也越来越高。然而，信息泄露等安全问题也应运而生，成为转账时不可避免的阻碍，亟待解决和改善。

1.1 课题的背景

随着我国的信息技术迅猛发展和国内互联网的发展和普及，网络的使用无处不在，并且与人们的生活息息相关，支付方式日新月异，转账等网上交易业务方便、快捷、安全、可靠、实时和易操作等技术成为热点。就以银行为例，网上银行越来越受大家欢迎，网络上的交易也越来越方便快捷，其中线上的转账业务给人们带来了便利和简单的操作，未来的生活离不开网络化和数字化，货币交易也离不开线上支付转账。社会在进步，人们也跟上了步子，纷纷加入了网银队伍，网银给广大消费者带来了极大便利，网银让人们不必到跑到银行柜台就能自己手动转账。

但是，线上交易越受欢迎，信息泄露和交易安全就越受关注。网上银行转账必需在输入卡号与密码才能进行，相对于其他转账方式而言，部分转账业务安全加密程度较低，其安全保障措施也不够完善，容易导致个人信息落入不法分子手中，从而造成用户资金被盗。

网上转账主要有三方面安全隐患：网站本身的安全性，用户和银行之间传递信息的安全性和交易信息在存储时的安全性。一些网上银行在用户登录时存在一些安全隐患，可能造成信息泄露。还有，用户的安全意识薄弱也致使网上银行的安全性不足。目前，我国一些银行用户设置密码时意识薄弱，用一些生日等容易窃取的数字作为密码，也造成安全隐患。第三方面，用户信息在系统中存储时可能存在被攻击而造成信息流失和被盗用的问题。系统遭到外界不法分子的破坏，可能会造成系统数据库里的数据外泄，使得这些不法分子能够得到用户的信息，从而威胁用户的人身安全和财产安全。倘若某些用户的信息泄露，不法分子可以通过数据库中流出的信息登录系统进行转账造成财产流失隐患，还可以通过流出的账单信息获取该用户的资金往来，给用户带来人身安全的隐患。

本系统从信息的存储安全这一方面考虑网上转账系统的安全方面的问题，解决了网上转账安全第三方面的安全隐患，用户信息在系统中存储时可能存在被攻击而造成信息流失和被盗用的问题。即使系统数据库因外界不良因素攻击导致数据库中的用户信息泄露，也使得不法分子无法直接通过数据库中存储的消息获取用户的相关信息，只能获得一些对他们无法理解和无法破解的密文信息串，从而降低了外界通过窃取数据库的信息直接获取用户信息的风险。

因此，以“转账系统的安全研究与应用实现”为题设计和实现了本系统。

1.2 课题的内容与意义

1.2.1 课题的内容

本课题通过研究转账服务的类型，过程及安全方法，和对于相关加密算法解密算法的研究与应用实现，研究分析 AES，RSA，MD5 加密算法的流程及每一步，每一次循环的机制原理，和调查分析电子货币转账的技术过程，在 Windows 环境下及 JSP，Eclipse，MySQL 开发工具下，采用 MVC 模式，设计并实现一个基于 AES，MD5 加密算法的小型模拟电子货币转账的系统。

AES（Advance Encryption Standard）加密算法，凭借自身算法结构简单，加密解密速度快，编码十分紧凑等优点，具有较高的开发潜力和较高的使用价值，被应用至信息保护，电子安全等方面。AES 这种加密算法是迭代运算的，密钥是对称的，即加密和解密密钥相同，每次都是以 4*4 的字节矩阵进行处理的，加密过程由轮密钥开始，进行 Nr 轮迭代运算。

MD5（全称 MD5 Message-Digest Algorithm）信息摘要算法，主要用于身份认证，是一种十分具有代表性的哈希算法。该算法处理信息的过程都是以 512bit 进行分组的，每 512bit 的数据又分为 16 个 32bit 的子分组，然后进一系列迭代运算，最后输出的是 4 个 32bit 的分组，将此级联后得到 128bit 的散列值，即为加密后的密文。

本系统的功能结构根据 MVC 模式框架进行设计和开发的。模型-视图-控制器（全称 Model-View-Controller，简称 MVC），就是依据输入输出和处理流程将整个系统结构分为 View，Model 和 Controller 层。视图（View）层用于与用户交互动作，可以用 HTML 和 JSP 实现。模型（Model）层主要用 Javabean 对业务逻辑和数据进行封装。控制器（Controller）层完成流程控制，接收 View 层用户的请求，选择合适的模型进行处理，最后根据处理结果选择合适的视图响应用户的请求，可以用 Servlet 实现。本系统的前后端设计都是基于 MVC 模式，实现前端与后端分离，避免设计页面时需要考虑到算法方面的问题，也避免了设计算法时需要考虑页面设计问题。MVC 通过中间的 servlet 组件起到控制器的作用，将前端输入信息数据作为参数传递到后端进行处理，将处理完的信息返回至前端进行显示，其中，MVC 发挥着十分重要的作用，也给系统的开发带来了便利。

本系统主要实现的功能模块包括：

注册和登录模块：查询 MySQL 数据库中的账户信息表，若注册的 ID 不重复以及其他信息格式满足要求，可通过前端页面注册账户，并将此信息（其中密码用 MD5 加密算法加密）添加到账户信息表中。登录时查询该表中的账号及密码是否匹配，数据库中是否存在该账户的信息数据，判断是否登录成功，以便进行更多操作，根据结果导入不同的操作页面。

信息查询模块：根据 JSP 页面上的不同热键响应相关的请求，将查询操作传递到后端，通过数据库 SQL 语句的查询操作，将查询后的信息通过中间组件控制器返回至前端，通过 JSP 页面显示给用户，从而实现与用户的交互。

转账模块：用户进行转账业务操作时，通过 JSP 页面填写转账信息，包括收款账号，收款人姓名，转账金额，备注等。系统默认该登录账户为账单的付款账号，系统在用户提交账单时，对以上信息分别通过 AES 加密算法，每 16 字节分为 4*4 的矩阵进行加密，生成的密文传输到 MySQL 数据库存储，实现转账信息的保密，从而保障转账系统的安全，同时增减相应账户的金额，完成转账操作。

历史账单查询模块：用户在完成账户登录操作之后，进行查询历史账单时，会对MySQL 数据库中的账单信息表进行查询操作，由于数据库中账单信息表存储的是加密后的信息，因此，查询时需要对该表中的密文进行 AES 解密，然后通过 SQL 语句匹配付款账号及收款账号是否为该登录账号，将查询后的结果返回至前端，显示给该登录账户，从而完成查询历史账单操作。

密码修改模块：用户修改密码时，要填写旧密码，新密码和确认密码，满足条件之后，系统对填写的新密码进行 MD5 加密，然后更新至用户信息表，下次登录时用本次修改的新密码进行登录。

系统模块结构图如图 1-1 所示。

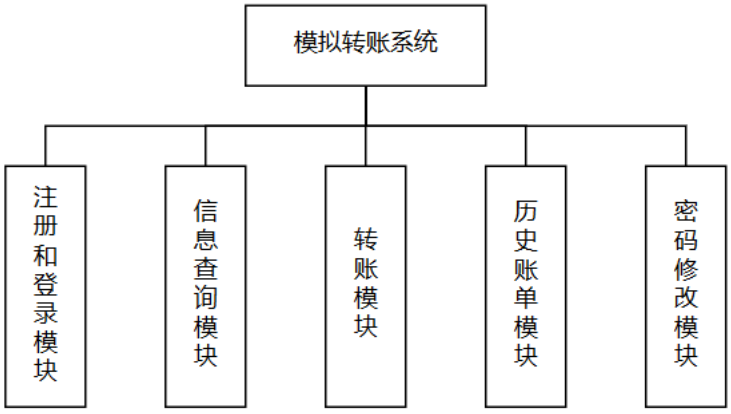


图 1-1　系统模块结构图

1.2.2 课题的意义

本课题的研究意义在于加强线上转账业务的安全和信息保护，能够在交易之余不会出现信息泄露，信息贩卖等问题，结合 AES，MD5 加密算法等技术，设计并实现一个模拟电子货币转账的系统。转账信息生成的同时，系统对转账信息进行加密，查询转账信息的时候在对其进行解密。对信息进行加解密不仅降低了用户信息的泄露，而且提高了转账业务的安全标准，进一步减少了因信息泄露带来的困扰，解决了用户的后顾之忧，从而具有广阔的应用、研发前景。

第二章　需求分析与可行性研究

本章节主要是针对系统的需求分析与可行性分析、相关技术的详细介绍。

2.1 可行性分析

随着互联网的高速发展，网上转账这一业务成为了社会生产，人民生活中不可或缺的一部分。同时，转账安全也伴随网上交易的流行而引起人们的关注，使得人们在线上转账时总是担心自己的信息产生泄露，导致自身人身安全和财产安全受到威胁。

本系统是模拟电子货币转账系统，可以进行线上转账及查询等功能，还出于对用户信息的保护这一方面做出改进，使得用户信息得到了进一步保障，降低了因信息泄露造成的损失。

2.1.1 项目要求

1、从功能方面来看：转账、查询用户信息、查询余额、查看历史账单等。

2、从性能方面来看：易操作、易上手、方便快捷。

3、从安全方面来看：信息加密、多重加密算法结合使用、安全可靠。

4、从技术方面来看：熟练掌握 Eclipse 及 MySQL 开发工具和 MVC 模式框架。

2.1.2 开发目标

1、用户可实现登陆及注册功能。

2、用户可以进行查询和修改自己的信息包括余额、用户信息等。

3、用户可通过此系统客户端进行转账操作，服务器（后台）进行信息加密存储。

4、此系统在用户提出查询历史账单请求时可提供解密后相关账单信息。

5、对三大类加密算法进行对比研究，选择合适的算法结合使用。

6、系统操作界面具有美观简单，交互性好，易操作等优点。

2.1.3 系统可行性研究

目前，人们的经济指标不断增长，转账信息安全问题发展成为困扰网上用户的一大威胁。可是，随着互联网的高速发展，网上转账这一业务成为了社会生产，人民生活中不可或缺的一部分，所以，不得不从信息泄露这一方面考虑转账系统的安全保障。

针对用户信息在系统中存储时可能存在被攻击而造成信息流失和被盗用的问题，本系统提出并实现将用户信息进行多重加密存储这一方案。即使系统数据库因外界不良因素攻击导致数据库中的用户信息泄露，也使得不法分子无法直接通过数据库中存储的消息获取用户的相关信息，只能获得一些对他们无法理解和无法破解的密文信息串，从而降低了外界通过窃取数据库的信息直接获取用户信息的风险，在系统设计与实现中起到了贯穿的作业。

2.1.4 技术的可行性研究

在技术研究方面,本系统是在 Windows10 环境下以及 Eclipse,MySQL 等开发工具的基础上，采用 MVC 模式,使用 Java 编程语言设计和实现的模拟转账系统。通过对三大类加密算法的研究，

有了多重加密算法的支持，使得用户能够在 web 网页上进行线上转账，实现模拟转账业务，又保障信息的保密，解决了用户信息泄露等不安全因素的问题，对线上转账的安全方面有十分重要的意义。

2.2 需求分析

2.2.1 市场需求分析

我国的网络化迅速覆盖和普及，人们的支付方式也日新月异，线上转账等网上交易的业务方便，快捷，安全，可靠，实时和易操作等技术成为了热点。可是，转账安全这一问题严重制约着信息网络化的发展，给网民便利的同时带来了一层阴霾，也是系统不可避免亟待解决的重点之一。

通过研究分析发现，影响网上转账安全问题的主要因素包括：[3]

1、网站本身的安全性：一些网上银行在用户登录时存在一些安全隐患，可能造成信息泄露。

2、用户和银行之间传递信息的安全性：在网上银行中，用户登录时的账号信息可能会被其他不法分子截取，非法登录盗取账户信息，造成财产被窃取。同时，用户的安全意识也影响着系统的安全性。

3、交易信息在存储时的安全性：用户信息在系统中存储时可能存在被攻击而造成信息流失和被盗用的问题。系统遭到外界不法分子的破坏，可能会造成系统数据库里的数据外泄，使得这些不法分子能够得到用户的信息，从而威胁用户的人身安全和财产安全。

本系统从信息的存储安全这一角度减轻了因信息泄露带来的威胁，解决了网上转账安全第三方面的安全隐患，降低了外界通过窃取数据库的信息直接获取用户信息的风险。

2.2.2 软件功能的设计目标与需求分析

结合以上市场需求，设计和实现系统运行总模块。

用户进入转账系统，使用账户账号和密码进行登录操作，view 层将输入控件的内容作为参数通过 control 层 servlet 组件传递给 model 层，model 层连接 MySQL 数据库，将传来的参数请求至数据库 account 表中查询，将查询结果再返回至 view 层，根据返回结果进行下一步操作。如果用户没有该转账系统的账号，进行注册操作，将注册 JSP 页面的信息作为数据通过以上 MVC 模式插入至 account 表中。

用户登录成功后进入业务选择页面，选择相应业务进行操作。

查询账户信息，通过调用 account 表中信息并返回显示，用户仍可选择修改账户信息，对个人信息进行修改，将修改后的信息存入数据库中，用户亦可仅查询账户余额。用户查看历史账单时，需要对 MySQL 数据库中 bill 表中的密文信息进行 AES 解密，然后通过 SQL 语句匹配到付款账号及收款账号是否为该登录账号，将匹配后的明文结果按照时间先后顺序返回至 view 层，显示给该登录账户。用户在进行转账核心业务时，填写账单并提交后，系统根据付款账号和收款账号进行相应账户余额变动，同时将转账各项信息进行 AES 加密，将密文信息存储至 MySQL 数据库 bill 表中，实现账单信息保密。

用户在完成某一业务后，返回至业务选择页面，仍可进行其他业务操作，或者退出系统。

系统核心模块流程图如图 2-1 所示。

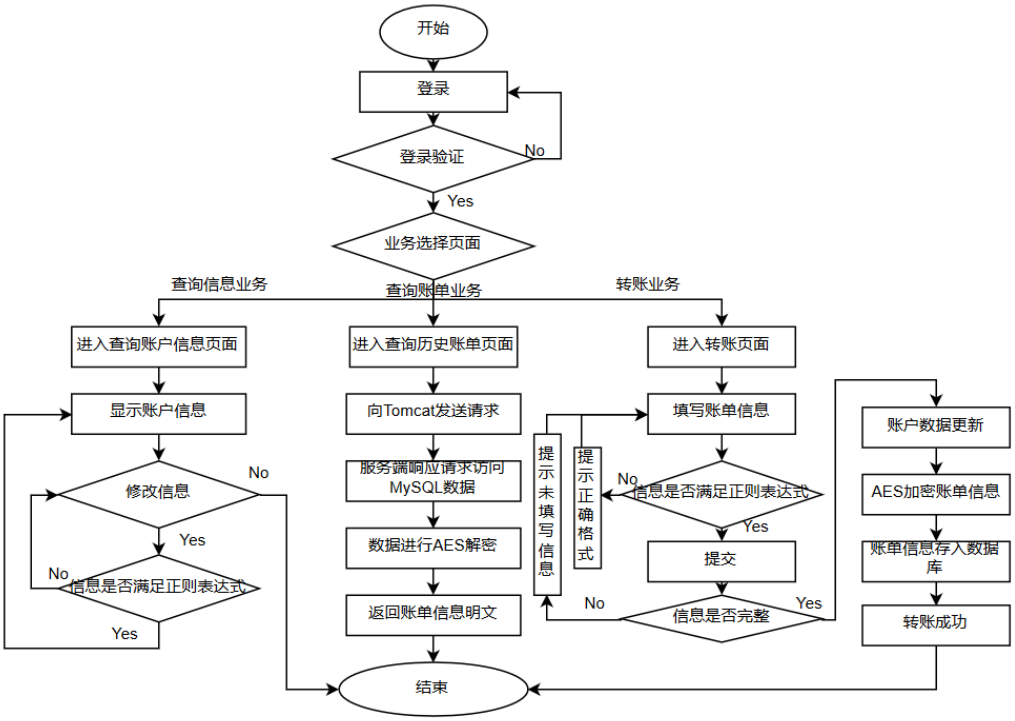


图 2-1　系统核心模块技术流程图

2.3 设计开发流程

典型的系统软件开发过程有瀑布式、增量式、团队软件过程、螺旋式等。本系统开发选择的是瀑布式软件开发过程，自上而下，在完成上一步的前提下开展下一步的工作，将前一项工作的结果传递给下一项的输入，以此类推完成整个模拟转账系统的开发流程。[4]设计开发流程如图 2-2。

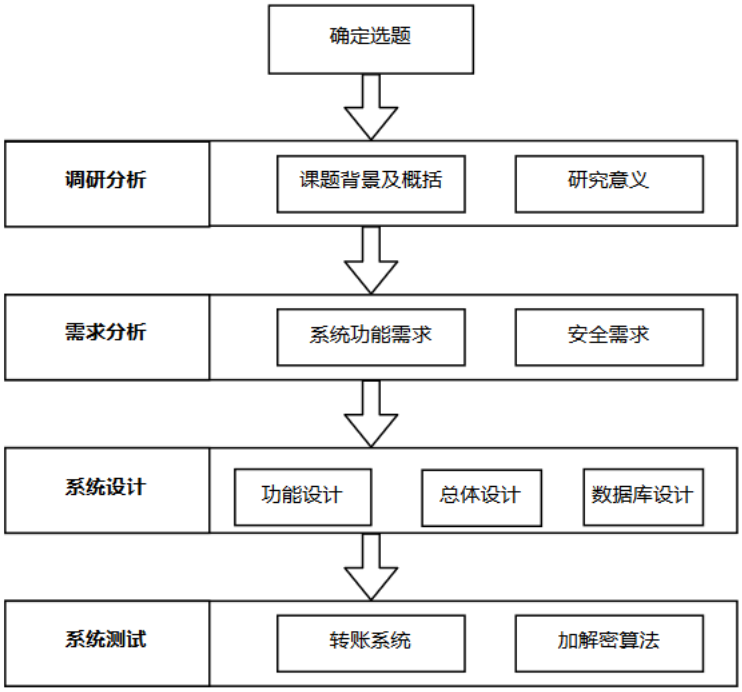


图 2-2　系统设计开发流程

采用瀑布式软件开发过程，首先进行市场需求分析调研，查阅资料分析网银安全问题以及信息泄露存在的问题和相应的解决方法。针对某一安全防护方面，对各类加密算法进行分析研究和对比，选择适合本转账系统使用的加密算法进一步学习调研。其次，划分系统功能模块，设计出合理的系统功能和结构，在 Eclipse 开发工具的辅助下，熟悉连接 MySQL 数据库，设计相应的数据库表，采用 MVC 框架完成各个功能模块的业务逻辑以及 UI 界面的设计。最后，对系统各个功能模块进行详细的测试，优化界面，使得整体简洁明了，方便快捷，容易操作，完善系统功能。

2.4 关键技术

2.4.1 JSP+MVC 模式+MySQL 数据库

本系统的前后端设计都是基于 MVC 模式，实现前端与后端分离，避免设计页面时需要考虑到算法方面的问题，也避免了设计算法时需要考虑页面设计问题。MVC 通过中间的 servlet 组件起到控制器的作用，将前端输入信息数据作为参数传递到后端进行处理，通过 JDBC 的操作存入或者读取出 MySQL 数据库中的信息，将处理后的信息返回至前端进行显示，其中，MVC 发挥着十分重要的作用，也给系统的开发带来了便利，所以以上模式比较切合本系统的设计需求。[7]系统模式设计如图 2-3。

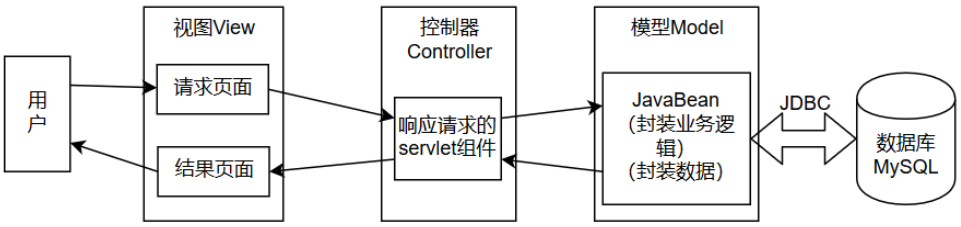


图 2-3　MVC 模式架构

1、MVC 模式

模型-视图-控制器，就是依据输入输出和处理的流程将系统结构分为 View，Model 和 Controller 层。视图(View)层用于与用户交互动作，可以用 HTML 和 JSP 实现。模型(Model)层主要用 Javabean 对业务逻辑和数据进行封装。控制器（Controller）层在中间其控制作用，接收 View 层用户的请求，选择合适的模型进行处理，最后根据处理结果选择合适的视图响应用户的请求，可以用 Servlet 实现。

2、MySQL 数据库

MySQL 数据库是一种开源的关系型数据库，常用 SQL 语句这类结构型语言进行管理。因为 MySQL 快速、可靠和适用广等优点，所以使用得比较广泛和普遍。

本系统通过 MySQL 数据库进行数据存储，主要包括账户信息和账单信息。账户信息包括账号，密码，余额，用户 ID，用户姓名，性别，电话号码，邮箱，联系地址等；账单信息包账单号，付款账号，收款账号，收款人姓名，转账金额，备注等。MySQL 支持本系统后端的数据处理，完成前端传输过来的数据操作请求。

3、JSP

Java 服务器页面（JavaServer Pages，简称 JSP）是一种动态网页技术标准。JSP 技术以 Java 作为脚本语言，能够为用户的 HTTP 请求提供服务，并且能与服务器上的其它 Java 程序一起处理复杂的业务需求。JSP 可以方便地添加动态网页内容，还继承了 Java 语言的优点，实现“一次编写，处处运行”的优势。

本系统的前端设计主要是通过 JSP 来完成，HTML 辅助，实现与用户进行互动，响应用户的请求，并接收用户提交的信息，通过中间的控制组件传递给后端，而且，经由后端处理之后的数据传递给前端，设计 JSP 页面完成用户请求返回结果的显示。

2.4.2 JavaScript

JavaScript（简称 JS）是一种基于对象和事件驱动的脚本语言，通过鼠标或热键的动作来进行事件驱动。[1]本系统采用 JS 主要也是 JS 来实现客户端输入时对输入信息格式以及内容的验证，只有验证成功之后，用户所输入的信息才能够传入后端，从而完成输入信息的预处理。当后端信息处理后传递给前端时，也是可以通过 JS 来控制信息的显示格式，从而使得用户能够更加直观地获取信息，所以比较切合本系统的设计需求。[2]

第三章　系统相关算法研究与实现

本章主要介绍在调研转账系统安全研究方面时所考虑的一些安全算法，主要是几种信息的加密算法，针对所学习了解的这几种加密算法，对其进行比较，分析每种算法的优势及特点，选择一些适合本系统的加密算法运用至信息加密中，主要是 AES，RSA，MD5 这三种极具代表性的加密算法。

3.1 AES 算法

AES 加密算法（全称 Advance Encryption Standard），由于存在诸多优点，如自身算法结构简单，加密解密速度迅速和编码紧凑等，具有很大的开发潜力和较高的使用价值，被应用至信息保护，电子安全等方面及领域。

AES 中的操作均是以字节作为基础进行操作的,每次都是以 4*4 的字节矩阵进行处理的，加密过程由轮密钥开始，进行 Nr 轮迭代运算。该算法用 Nr 表示对数据分组加密的轮数。其中，该算法的密钥长度有三种：16byte、24byte 和 32byte，不同密钥长度对应着不同加密轮数和分组长度。由于本系统加密的信息字段不是很长,所以选择的是 16byte 的密钥长度，该密钥长度对应的信息分组长度比较切合本系统的设计需求。具体对应关系如表所示。

表 3-1　AES 中 Nk,Nb 与 Nr 的对应关系

加密标准	密钥长度 Nk	分组长度 Nb	加密轮数 Nr
128	4	4	10
192	6	4	12
256	8	4	14

3.1.1 AES 加密流程

AES 这种加密算法的密钥是对称的，即加密和解密密钥相同，每次都是以 4*4 的字节矩阵进行处理的，加密过程由轮密钥开始，进行 Nr 轮迭代运算。[5][10]具体的 AES 加密流程图如图 3-1。

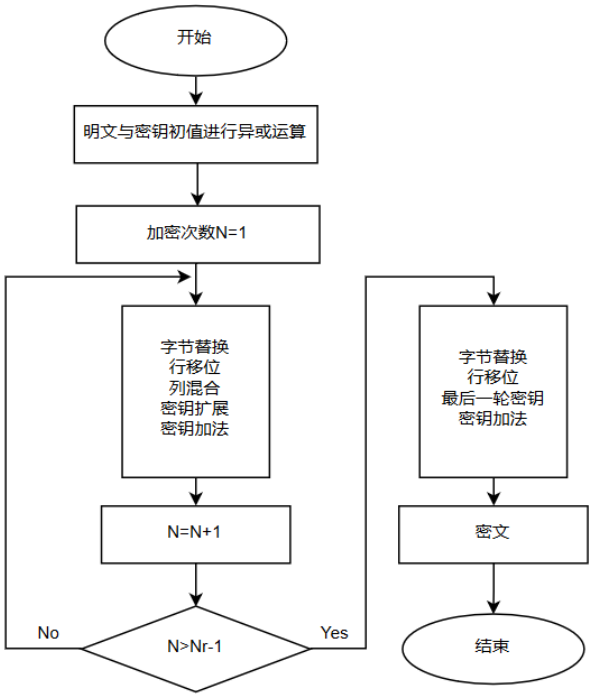


图 3-1　AES 加密流程

1、字节替换 SubBytes

AES 加密算法定义了一个 S 盒，通过 S 盒可以将 State 矩阵每个字节映射成相应的字节，映射规则为该字节的高 4 位为行值，低 4 位为列值，找到 S 盒中元素进行替换。例如，十六进制数 {24}。对应 S 盒的行是 2，列是 4，相应的 S 盒中该位置对应的值是 {2c}。

其实，构造 S 盒进行字节替换只是 AES 第一步进行的一系列数学运算的简化与规律总结，其中包括了取逆运算和仿射变换运算。只是因为这两种运算都是输入和输出相互一一对应的，不同的输入分别对应不同的输出，相同的输入对应相同的输出。而且 8 位最多表示 256 种代换，S 盒囊括全部，所以通过存储 S 盒来简化字节替换。取逆运算是基于有限域 $GF_2^8 = Z_2[x]/(x^8 + x^4 + x^3 + x + 1)$的逆。仿射变换是将取逆运算所得和多项式进行多项式乘法，再与一个常量进行异或运算，最后得到的结果与 S 盒替换所得的结果相同。仿射运算如下图 3-2。

$$\begin{bmatrix} b_7 \\ b_6 \\ b_5 \\ b_4 \\ b_3 \\ b_2 \\ b_1 \\ b_0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} a_7 \\ a_6 \\ a_5 \\ a_4 \\ a_3 \\ a_2 \\ a_1 \\ a_0 \end{bmatrix} \oplus \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

图 3-2　仿射运算公式

2、行移位变换 ShiftRows

行移位变换就是每一行根据相应规则进行循环左移，第一行不变，第二行移动一个字

节，第三行移动两个字节，第四行移动三个字节。具体如图 3-3 所示。

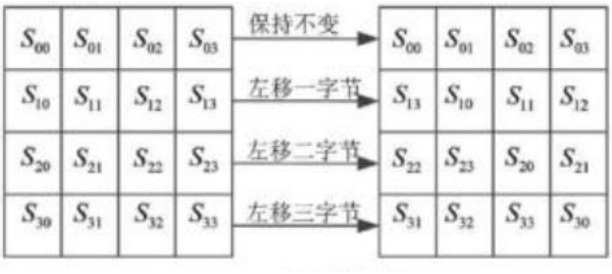


图 3-3　行移位变换

3、列混合变换 MixColumns

列混合变换将 State 矩阵每一列看作一个向量进行运算，对 x^4+1 取模，和多项式 a(x)做乘法运算,列混合运算公式如图 3-4。

$$\begin{bmatrix} s'_0 \\ s'_1 \\ s'_2 \\ s'_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix}$$

图 3-4　列混合变换

4、轮密钥加 AddRoundKey

将上一步列混合得到状态矩阵与这一轮相应的扩展密钥矩阵进行逐位异或，输出密文矩阵即为最后加密结果。[8][9]轮密钥加操作具体流程如图 3-5 所示。

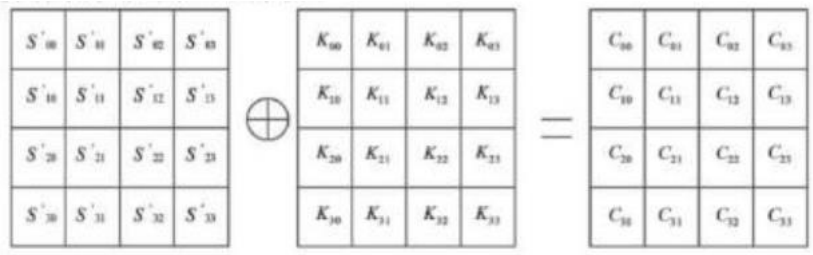


图 3-5　轮密钥加操作

3.1.2 AES 解密流程

由于 AES 算法是是极具代表性的对称加密算法，因此，AES 解密过程和加密过程相关，两者互为逆过程，且加密时密钥和解密时密钥相同。不过，解密开始时需要将密文先与最后一轮密钥进行异或操作。AES 解密算法流程图如图 3-6。

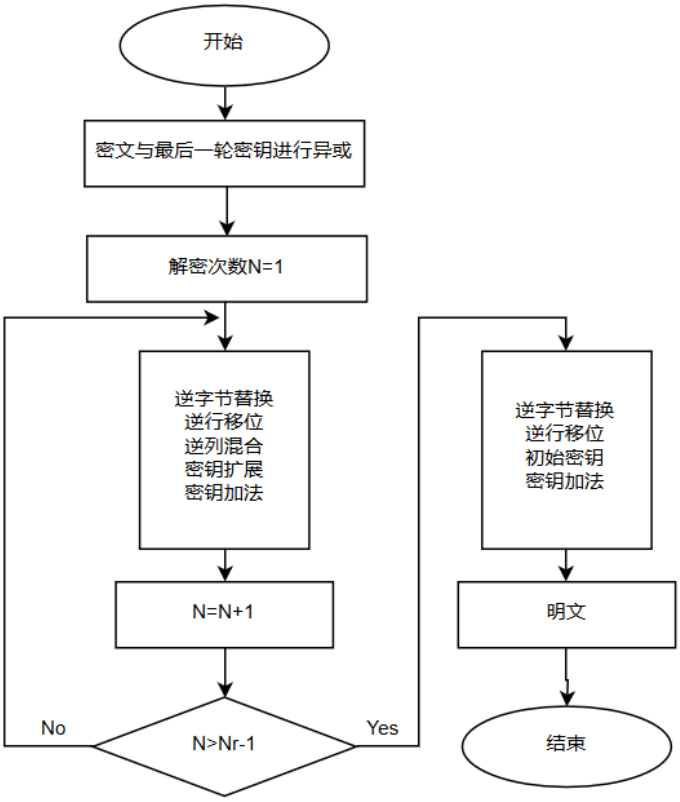


图 3-6　AES 解密流程

解密算法循环中具体各操作都是加密算法中的逆操作，就不再详细论述了。

3.2 RSA 算法

RSA 加密属于非对称加密中比较典型的一种，使用者可以在不直接传递密钥的情况下，通过原先分配的密钥进行解密。这样提前约定好密钥对即公钥与私钥能够确保信息的安全性，降低了直接传递密钥致使泄露带来加密信息被破解的风险。

这种加密模式可以表示为：AB 双方需要传递消息但不希望其他人能够获得消息，AB 需要提前约定好一对密钥对包括公钥和私钥，公钥公开，私钥私钥自己保留，A 给 B 发消息，A 用公开的公钥加密，B 收到密文信息后，用自己保留的私钥进行解密即可获得明文消息。

对于密钥对生成过程而言，涉及到一些数学方面的函数运算，需要两个素数 p 和 q，并计算出 n = pq 的结果，找到满足(d,φ(n))的表达式，就是找出满足 d 与 φ(n)互质的组合，其中需要满足 0 < d < φ(b)且 φ(n)是欧拉函数。由于 n = pq，φ(n)=(p-1)(q-1)，进一步说是 d 与(p-1)(q-1)互质，再找出 0 < e < φ(n)满足 de ≡ 1(mod φ(n))表达式，使得 de 除以 φ(n)余数为 1，从而分别产生了密钥对(n,d)和(n,e)。该算法使用秘钥对(n,x)加密的方法为 f(a,n,x) = ax(mod n)，解密仍然为该函数,这样一来就可以很容易实现用公钥加密私钥解密或用私钥加密公钥解密这一操作，即为 B = f(A,n,d)且 A = f(B,n,e)A=f(B,n,e)。

3.3 MD5 算法

MD5（Message-digest Algorithm 5）信息-摘要算法，是一种极具代表性的哈希算法，主要特点是其单方向性，你可以将明文信息通过 MD5 加密成密文信息，但是无法通过解密密文来获取明文信息。[11]

在 MD5 加密算法中，第一步对信息进行填充，使其字节长度达到(N+1)*512 的要求。其次，有 4 个 32bit 被叫做链接变量的 A，B，C，D，这四个变量初值赋值给 a，b，c，d 好了后，进行 N+1 次大循环，每一次大循环中包括四轮迭代运算，每一轮又包括 16 次不同参数的运算，每次运算都是对 a、b、c、d 中的 3 个进行非线性函数运算，将得到的结果加进行加法运算，加数包括未参与运算的第 4 个变量，512bit 明文的某个 32bit 分组和一个常数。再将所得结果向右环移一个不定的数，再加上 a，b，c，d 中之一，将此结果赋值给 a，b，c，d 中之一。在四轮循环结束后，将 A，B，C，D 分别加上 a，b，c，d，再用下一组 512bit 明文继续运行该算法，直至最后输出的是 A，B，C，D 的级联，所以无论明文多长，所得密文都是 128bit。具体加密流程图如图 3-7。

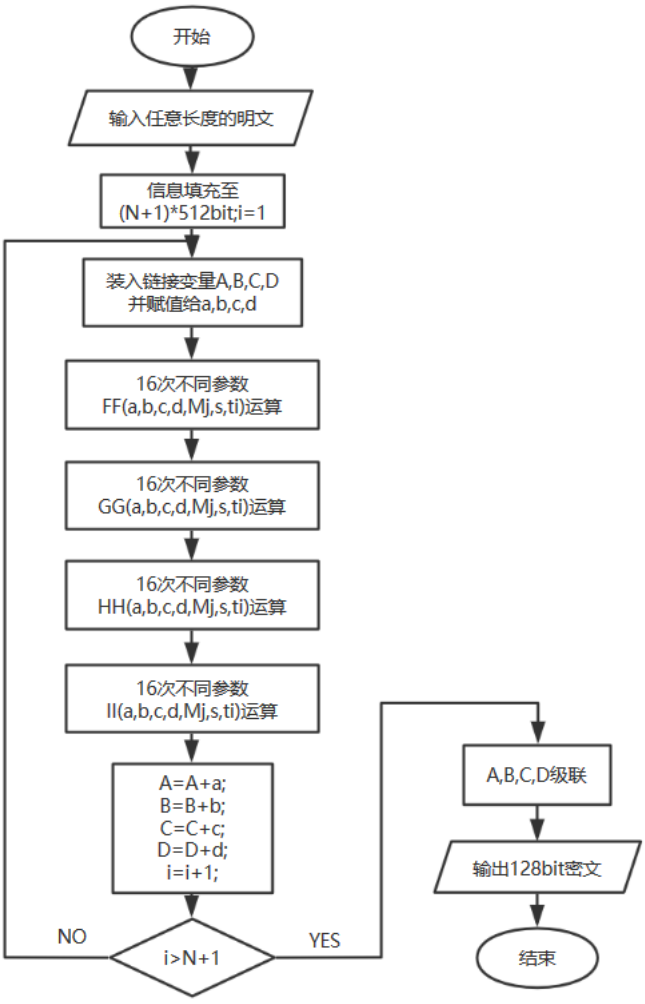


图 3-7　MD5 加密流程

3.4 加密技术的选择与结合

3.4.1 各类加密技术比较分析

加密技术主要分为三大类：对称加密，非对称加密和散列（哈希）算法加密。这次针对于转账系统安全研究方面对三大类加密技术进行了调研和学习，分别选择了三类加密技术最具代表性的进行比较分析，其中有对称加密中的 AES，非对称加密中的 RSA，哈希算法中的 MD5。

AES 加密和解密使用的密钥是同一个密钥，客户端和服务器端双方都需要知道加密算法和密钥，所以对于密钥的保存至关重要。AES 有一些十分明显的优点，速度快，安全级别高，面向字节加密，高效，适合大量数据加密。但是存在密钥泄露而引起信息泄露的问题。

RSA 作为非对称加密技术的代表之一，它支持密钥变长，加密文件块变长。公钥加密，私钥解密，安全系数比对称加密高，但是加密速度比对称加密慢。而且对于密钥对即公钥和私钥的管理带来了不便，增加了密钥对的生成和分配，存储管理的负担。

MD5 作为不可逆的加密算法，只能加密，不能解密，而且目前为止技术上逆运算的程序并没有被开发出来。因此，这类算法对于信息传递来说基本没有什么用处，但是，正因为它的不可逆性，使得它十分适合用于身份认证与数字签名这方面，可以防止信息被修改。而且 MD5 有一个极大的优势，它不会随着明文长度的增加导致密文长度随之增加，无论需要加密的信息多长，加密之后是一段唯一的固定长度 128bit 的密文字符。所以，MD5 非常适合用于本转账系统登录时的身份验证。

3.4.2 加密算法选择与结合

本系统处于信息安全的考虑，在因信息泄露带来损失的情况下，选择给系统业务增加多种加密算法，使得用户的信息得到多重保护，即使被外界攻击导致数据库内信息外泄，针对数据库里的用户隐私信息也无法直接得到明文信息。出于对上述三种典型加密技术的考量，根据本系统的需求，选择如下方案。采用 MD5 和 AES 相结合的方案，充分发挥 MD5 在身份验证上的优势以及 AES 在账单信息加密上的高效性，大数据量的处理能力和密钥的易管理等优点。

用户注册账户时，填写密码提交表单成功后，后端对提交的密码进行 MD5 加密，将加密后的密码信息存储到 MySQL 数据表中，当用户登录时，同样的方式进行加密，将输入密码加密后的信息与数据库中的信息进行比较，匹配成功后即可完成身份验证，成功登录本系统。在用户进行转账时，提交表单后，后台自动调取用户加密后的密码信息作为 AES 加密的密钥，同时针对需要加密的各个字段分别进行 AES 加密，将加密后账单信息存入数据库中。当用户提交查询历史账单请求时，后台自动调取用户加密后的密码信息作为 AES 解密的密钥，通过 SQL 语句查询匹配到需要的账单信息后进行 AES 解密，将加密后的账单信息返回给用户，从而达到及时响应用户请求又保障用户信息安全存储的效果。

第四章　系统功能设计

本章节是对系统的整体功能模块分析与设计及数据库设计的介绍。

4.1 系统分析与设计

4.1.1 整体功能模块分析

经过调研分析，本系统研究转账服务的类型、过程及安全方法以及转账安全算法的研究和实现，设计一个基于 AES 的模拟电子货币转账的系统程序，客户端主要功能模块包括：账户注册，账户登录，查询余额，转账业务，查询或修改账户信息，查询历史账单，修改密码等。其中转账业务包括通过 AES 加密算法对账单信息进行加密和通过 AES 解密算法对加密后的账单进行解密和通过 MD5 进行用户身份认证。

各个功能模块如下：

1、账户注册及登录模块：该模块主要负责新增用户的处理以及用户登录时的身份验证，其中包含通过 MD5 来进行身份认证。

2、查询余额模块：该模块主要是查询数据库中的余额字段。

3、用户信息模块：该模块主要包括查询用户信息以及修改用户信息。

4、转账模块：该模块是本系统的核心模块，主要是负责转账业务，其中包含 AES 加密账单信息。

5、查询历史账单模块：该模块也是本系统的核心模块，主要是加密信息在数据库中检索和输出，包含 AES 解密。

6、修改密码模块：这部分是负责密码的修改和加密信息后更新。

7、注销模块：该模块即为退出系统功能。

系统设计功能模块图如下图 4-1 所示。

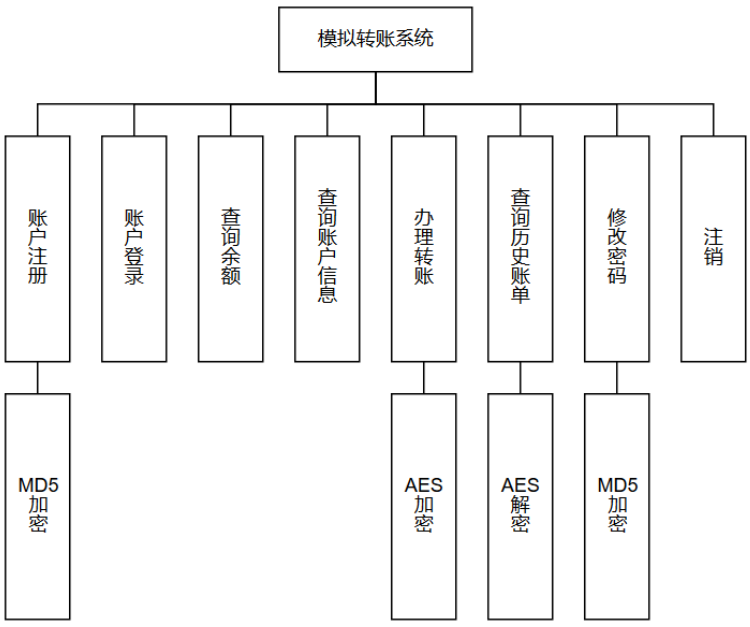


图 4-1　系统设计功能模块图

4.1.2 账户注册及登录模块

用户注册账户时，查询 MySQL 数据库中的账户信息表，若注册的 ID 不重复以及其他信息格式满足要求并且提交至后端，系统会将账户密码进行 MD5 加密，产生 128bit 的密文字符串，并将此信息添加到账户信息表中。登录时查询该表中的账号及加密后密码是否匹配，数据库中是否存在该账户的信息，判断是否登录成功，以便进行下一步操作。该模块主要是用 MD5 算法来进行用户的身份认证。具体模块流程图如图 4-2 所示。

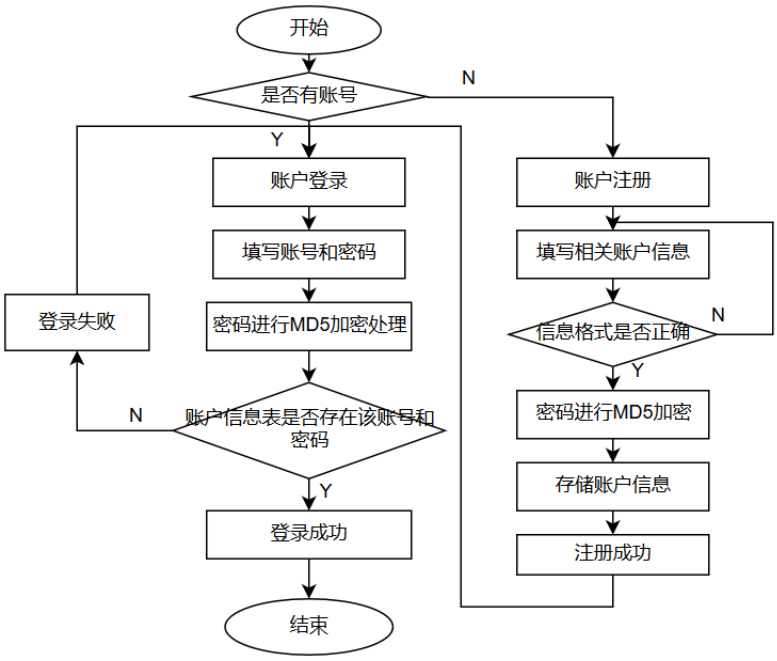


图 4-2　注册登录流程图

4.1.3 查询余额模块

客户端发送查询余额请求，传递给后端进行分析处理，检索数据库字段，将查询到的余额返回至前端，以及时响应客户端请求，让用户能够很快的获取账户余额信息。具体流程图如图 4-3 所示。

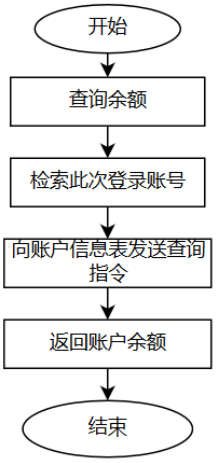


图 4-3　查询余额流程图

4.1.4 用户信息模块

根据 JSP 页面上的不同热键响应相关的请求，将查询操作传递到后端，通过数据库 SQL 语句的查询操作，将查询后的信息通过中间组件控制器返回至前端，通过 JSP 页面显示给用户，从而实现与用户的交互。该模块包括查看用户信息和修改用户信息功能。具体流程图如图 4-4 所示。

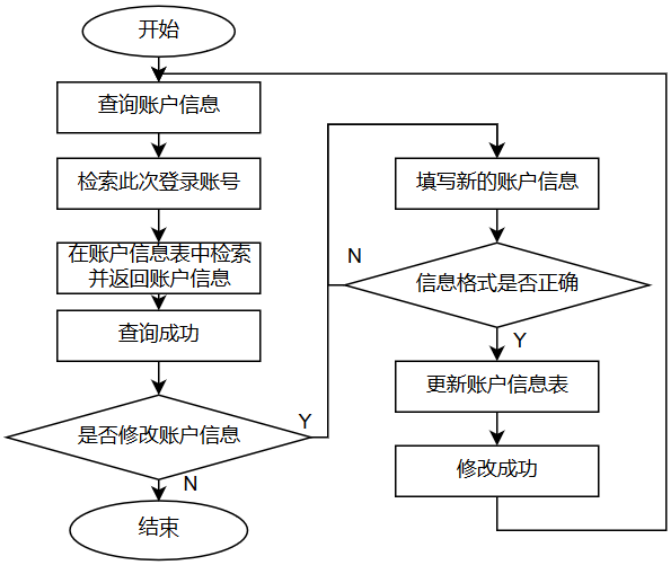


图 4-4　查询账户信息流程图

4.1.5 转账模块

用户办理转账业务时，填写转账信息提交后，系统根据输入信息判断信息是否合格，比如转账金额不能为负数，收款人账号和姓名相匹配，账户余额大于转账金额。提交的信息条件满足后传入后端，系统获取账户密码作为 AES 加密算法密钥的种子，对提交的账单

相关字段信息分别进行 AES 加密，其中包括加密信息的预处理和加密后信息的处理。将加密的账单信息存储到数据库相关数据表中，同时进行账户信息表余额的相应变化，实现账户之间的转账操作。具体流程图如图 4-5 所示。

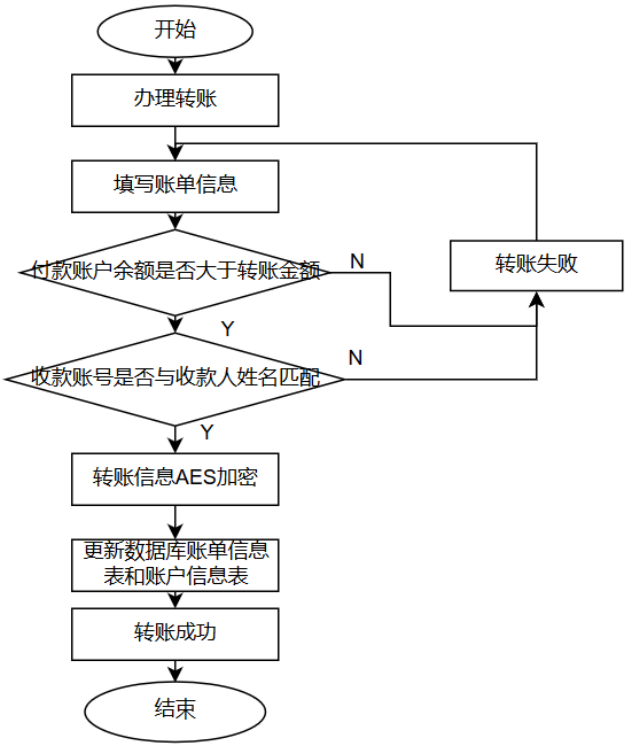


图 4-5　转账流程图

4.1.6 查询历史账单模块

用户在进行查询历史账单时，会对 MySQL 数据库中的账单信息表（包括账单号，付款账号，收款账号，收款人姓名，转账金额，备注等）进行查询操作，由于数据库中账单信息表存储的是加密后的信息，因此，查询时需要对该表中的密文进行 AES 解密，系统获取账户密码作为 AES 解密算法密钥的种子，然后通过 SQL 语句匹配，将查询后的结果返回至前端，显示给该登录账户，从而完成查询历史账单操作。具体流程图如图 4-6 所示。

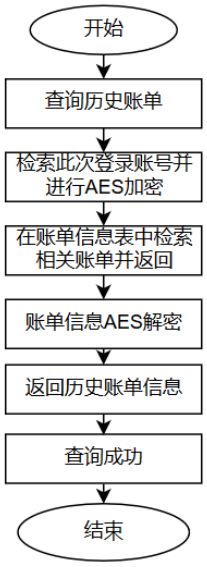


图 4-6　查询历史账单流程图

4.1.7 修改密码模块

用户执行修改密码操作时，需要输入旧密码和新密码和确认密码，当旧密码不符合数据库中存储的密码数据不一致时，提醒用户错误信息，在此基础上，当新密码和确认密码一致时，将新密码进行 MD5 加密成 128bit 密文去更新数据库中密码信息，下次用户登录即可使用该密码进行系统操作。具体流程如图 4-7 所示。

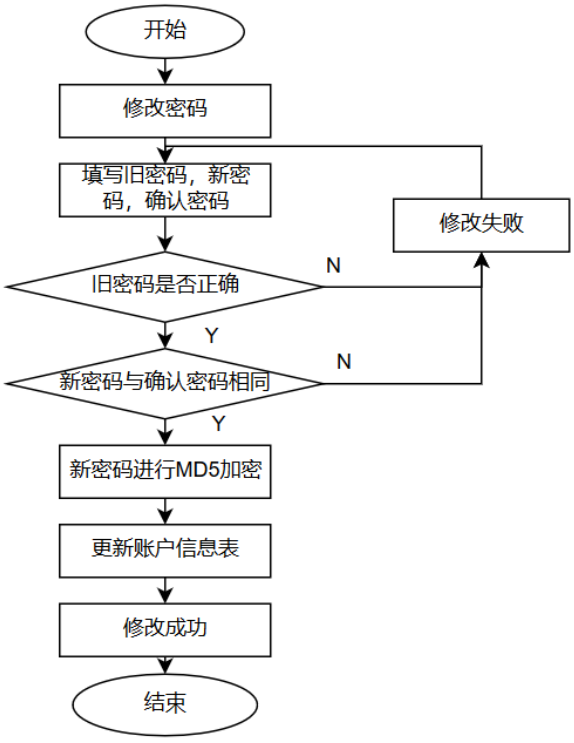


图 4-7　修改密码流程图

4.2 数据库设计

数据库是整个系统的关键部分，是整个用户信息存储的核心，存放着关系结构化信息。对于一个系统的数据库而言，信息存储安全，方便，快捷，稳定是衡量数据库性能的关键因素，因此，针对以上特点，设计出了符合本系统要求的数据库。

4.2.1 需求分析

经过对本模拟转账系统的研究分析，对本系统的数据库设计出以下内容。

账户信息：账号，密码，余额，用户名字，用户 ID，用户性别，用户电话，用户邮箱，用户地址。

账单信息：账单编号，付款账号，收款账号，收款人姓名，转账金额，转账日期，备注。

4.2.2 概念模型设计

根据以上数据库信息关系分析，设计实体-联系图如下。

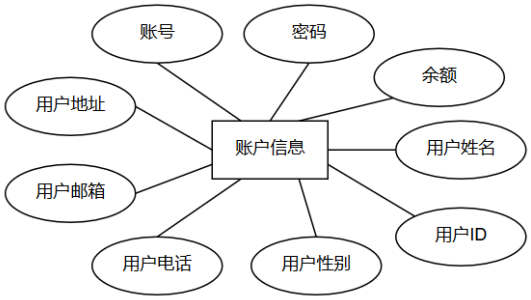


图 4-8　账户信息 E-R 图

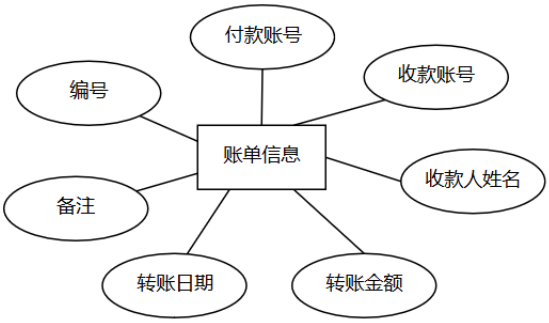


图 4-9　账单信息 E-R 图

4.2.3 数据库详细设计

数据库中三级模式中模式层将实体-联系图转化为相应的数据库表，具体的数据库表如下所示。

表 4-1 账户信息表

字段名	数据类型	长度	说明	描述
ano	varchar	16	Primary key	账户账号
apwd	varchar	15	Not null	账户密码
amoney	float		Not null,default ‘0’	账户余额
aname	varchar	16	Not null	用户名字
aId	varchar	18	Not null, unique	用户 Id
asex	varchar	2	Not null	用户性别
atel	varchar	11	Not null	用户电话
amail	varchar	30	Not null	用户邮箱
address	varchar	50	Not null	用户地址

账户信息表用于存储账户的信息，以便登录，注册，查询账户信息，查询余额等操作。

表 4-2 账单信息表

字段名	数据类型	长度	说明	描述
bno	int	10	Primary key,auto_increment	账单编号
boutno	varchar	16	Not null	付款账号
binno	varchar	16	Not null	收款账号
bname	varchar	16	Not null	收款人姓名
bmoney	varchar	16	Not null	转账金额
bdate	date		Not null	转账日期
bnote	varchar	64		备注

账单信息表用于存储账单的信息记录，用于转账，查询历史账单业务。

4.2.3 数据库的链接

用户每次进行业务操作时，都需要访问数据库，同时用户每次操作的结果也需要更新至数据库中，以保证数据的一致性。每一次用户操作都进行数据库的链接会造成大量的代码重复冗余，降低代码的可读性，因此，针对数据库的链接，将其相同代码封装至 model 层的 JDBCBean 包中，每次需要链接数据库时，只需调用此包，输入相关参数即可，大大降低了代码冗余。[6]代码如下：

```
public JDBCBean() {
    connStr="jdbc:mysql://localhost:3306/lk?useUnicode=true&characterEncodin
g=gb2312&useSSL=false&serverTimezone=UTC";
    try{Class.forName(driverStr);//导入驱动，加载具体的驱动类
        conn = DriverManager.getConnection(connStr,dbusername,dbpassword);
//与数据库建立连接
        stmt = conn.createStatement();//发送 sql，执行（增删改，查）
     }catch(Exception ex){System.out.println("无法同数据库建立连接！");
                ex.printStackTrace();}}
```

第五章　编码与实现

本章主要介绍了系统主要模块的编码实现和效果界面展示图。

5.1 主页的编码实现

5.1.1 主页功能

主页主要是各个功能的导航页面，通过各个链接索引至相应模块完成相应的操作。其中包括查询余额模块，查询账户信息模块，转账模块，查询历史账单模块，修改密码模块等。

5.1.2 实现方法

主页是各个功能的导航，将主页界面划分成 4 个部分，每次操作转换相应部分界面从而进行相应操作，其中主要是通过 frameset 控件进行界面划分，每个框架再通过 frame 控件来定义其内部结构，其中菜单栏中通过链接形式导航至相应操作界面。

主要代码如下：

```
    <frameset rows="10%,*,7%" frameborder="no" framespacing="0">
    <!--将界面上下布局分为三部分-->
    <frame src="head.jsp" frameborder="0" scrolling="no" ><!—主页的抬头-->
    <frameset cols="15%,*" frameborder="no" framespacing="0">
    <!—将该部分左右布局分为两部分-->
    <frame src="left.html" frameborder="0" scrolling="auto"> <!—左侧菜单-->
    <frame src="welcome.jsp" frameborder="0" name="showframe"
scrolling="auto" ><!—右侧显示用户操作的界面-->
    </frameset>
    <frame src="foot.html" frameborder="0" scrolling="no"><!—主页的页脚-->
    </frameset>
```

主页界面如图 5-1 所示。

图 5-1　主页界面

5.2 注册登录模块的编码实现

5.2.1 注册登录模块功能

注册就是用户在未拥有本系统的账户时通过填写表单注册，填写的信息满足系统要求即可，提交表单即可注册成功，反之，注册失败。

登录操作即需将输入的账号和密码与数据库中存储的账号和密码进行比对，符合即登陆成功，相反即失败。

5.2.2 实现方法

用户通过前端 JSP 页面填写并提交完整表单后，系统根据填写信息进行判断，符合要求即可通过后端处理注册信息，密码部分进行 MD5 加密，将处理后的信息增添至账户信息表中即注册成功。例如，用户需要注册一个新账户时，点击注册进入注册页 register.jsp，填写表单（账号：9087556421333330　密码：lklk11　确认密码：lklk11　姓名：罗缇　ID：123456789012345678　性别：男　电话：13423121177　邮箱：12343@qq.com　地址：新昌街 33 号），检测填写信息不为空之余，还需检测账号是否都为数字，密码和确认密码是否一致，ID 是否满足身份证号格式，电话号码是否满足电话格式正则式，邮箱地址是否满足对应格式的正则式，这些格式的检测合格之后，前端才会将这些信息传递至后端，控制层 AccountInfoController.java 通过 request.getParameter() 读取这些信息，创建一个新的 AccountInfo 对象，通过 set 函数赋值给相应的属性，其中，密码需要先进行 MD5 加密，加密后的密文再进行赋值，然后将这个对象添加至 Account 表中，添加成功返回相应标识，控制层根据返回的参数传递给前端，跳转至注册成功界面 register_success.html。

用户登录填写账号和密码，点击登录后，系统将读取前端传输过来的参数，密码部分进行MD5加密，将处理后的信息与账户信息表中的信息比对，在数据表中找到符合的信息即登陆成功，同时创建session。例如，某用户需要登录系统时，点击进入登录界面 login.jsp，输入信息（账号：1122334455667788　密码：123456），提交后，前端传递信

息至后端，控制层AccountInfoController.java通过request.getParameter()读取这些信息，同样，创建一个新的AccountInfo对象，通过set函数赋值给相应的属性，其中，密码需要先进行MD5加密，加密后的密文再进行赋值，然后，调用AccountInfo类中的login()函数，即在Account数据表中检索账号和密码是否对应存在，成功即返回成功标识，控制层根据返回的参数跳转至系统主页main.jsp。同时登陆成功时，系统通过request.getSession().setAttribute("ano",accountno)创建session，直至用户注销时通过request.getSession().removeAttribute("ano")和request.getSession().invalidate()注销session。

注册登录界面如图 5-2，5-3 所示。

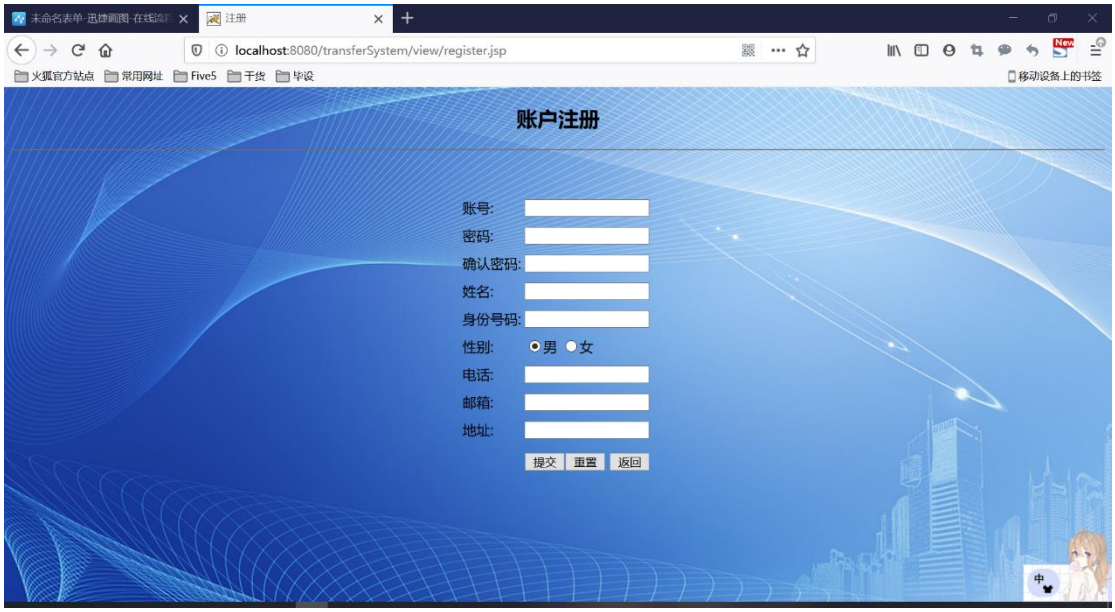


图 5-2　注册界面

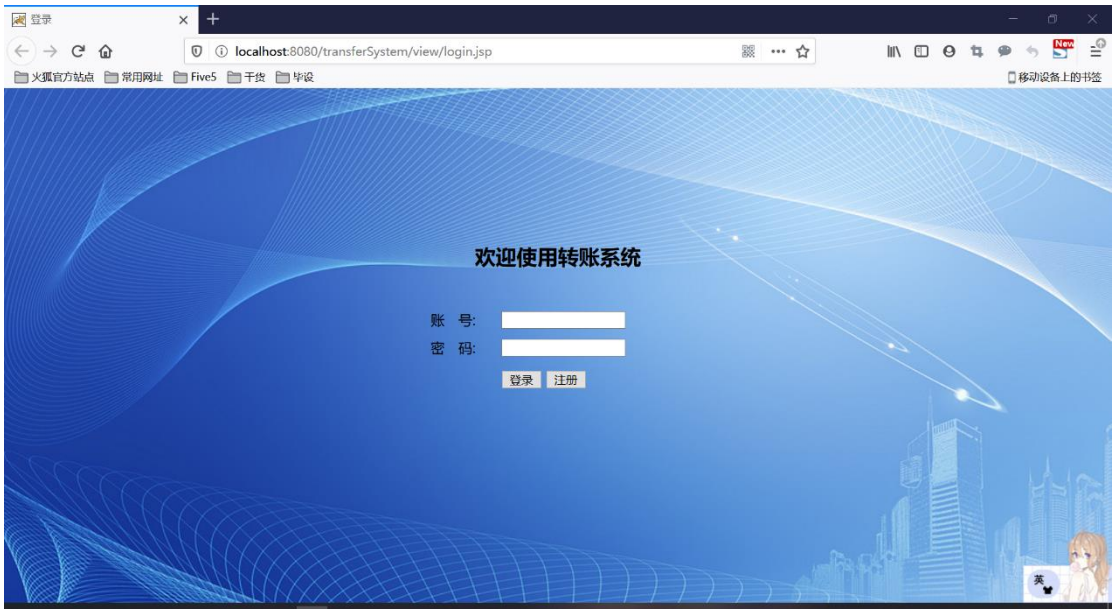


图 5-3　登录界面

5.3 查询账户信息模块的编码实现

5.3.1 查询账户信息模块功能

该模块主要是细分为查询余额和查看账户信息。用户选择此功能即可查询到本账户的相关信息。

5.3.2 实现方法

用户选择该功能时，后端接收到前端的请求，通过读取 session 中此次登录账号，在账户信息表中检索，找到匹配的信息返回至前端，通过相关控件规整地显示给用户。例如，用户点击查询账户信息后，控制层 AccountInfoController.java 接收到 query_accountinfo.action，读取 session 中的账号，创建一个新的 AccountInfo 对象，通过 setAno()函数赋值给相应的属性，调用 AccountInfo 类中的 getAccountInfo()函数，即检索 Account 表中相应数据字段，返回该账号的账户信息 list，传至控制层 AccountInfoController.java，最后传递至视图层 query_accountinfo.jsp 中显示出来。

查询账户信息界面如图 5-4 所示。

图 5-4　查询账户信息界面

5.4 转账模块的编码实现

5.4.1 转账模块功能

该模块主要是完成转账业务，成功将提交账单后，满足信息的格式要求后，对相应字段进行加密，存储到数据库表中和更新相应数据。

5.4.2 实现方法

当用户办理转账业务时，用户需要填写转账的表单，其中有收款账号，收款人姓名，转账金额，备注（选填）。系统在用户提交表单后进行判定，转账金额是否符合要求，收款账号和收款人姓名是否配对，账户余额是否大于转账金额等，符合系统要求之后，读取 session 中的账号作为付款账号，查询账户的密码作为 AES 加密密钥的种子，读取系统的

日期作为转账日期，结合用户填写的信息作为完整账单的信息。通过 AES 加密算法对付款账号，收款账号，收款人姓名，转账金额，备注加密，存入账单信息表中，同时，修改更新账户信息表付款账号和收款账号的账户余额信息。

例如，用户转账操作时，点击进入转账界面 transfer.jsp，填写表单（收款账号：1234567890123456　收款人姓名：罗帅　转账金额：55　备注：今天天气真好啊!），提交至后端，控制层 AccountInfoController.java 接收到 transfer.action，通过 request.getParameter()读取前端的参数，判断转账金额是否为负数以及是否为非数字，付款账户余额是否满足转账的金额，收款账号是否与收款人姓名相匹配，符合条件后，调用 AccountInfo 类中的 transfer_out()和 transfer_in()函数变更付款账户和收款账户的余额。同时，创建一个新的 BillInfo 对象，读取该账户的密码作为 AES 加密的密钥种子，调用 BillInfo 类中的 AES 加密函数 encrypt 函数对相应字段进行加密，将密文分别赋值给对应得 BillInfo 对象的属性中，然后将该对象添加到 Bill 表中。将上述三个操作的结果满足 result_out==1 && result_in==1 && result_tran==1 即为转账成功，跳转到转账成功界面 transfer_success.html，反之，跳转至转账失败界面 transfer_fail.jsp。

主要代码如下：

```
    int result_out=aout.transfer_out(bmoney);///修改付款账户余额
    int result_in=ain.transfer_in(bmoney);//修改收款账户余额
    String pwd = aout.Query_pwd();//读取付款账户的密码密文作为密钥种子
    BillInfo bi=new BillInfo();
    String en_boutno=bi.encrypt(boutno,pwd);//AES 加密相关信息字段
    bi.setBoutno(en_boutno);
    ……
    int result_tran=bi.transfer();//返回转账结果
    if(result_out==1 && result_in==1 && result_tran==1) {
        System.out.println("转账成功");
        request.getRequestDispatcher("/view/transfer_success.html").forward(
request, response);//转账成功跳转
     }
```

转账界面如图 5-5 所示。

图 5-5　转账界面

5.5 查询历史账单模块的编码实现

5.5.1 查询历史账单模块功能

该模块主要负责将其账号匹配的账单信息筛选出来，并将其解密得到的明文返回前端分页显示出来。

5.5.2 实现方法

用户查询历史账单时，后端接收到前端传来的请求，读取 session 中的登录账号，查询账户的密码作为 AES 加解密密钥的种子，将此登录账号进行 AES 加密，将此密文作为检索的信息字段，将检索到的结果通过 AES 解密，得到的明文信息添至 list 中，将其按转账日期排序，返回至前端分页显示给用户，所以用户能直接看到查询后的明文结果，又保障了信息的存储安全。

例如，用户进行查询历史账单请求时，控制层 AccountInfoController.java 接收到 query_billinfo.action，通过 request.getSession().getAttribute("ano")读取系统登录账号，创建一个 AccountInfo 对象，调用 Query_pwd()获取 AES 加解密密钥的种子，再创建一个 BillInfo 对象，调用 encrypt 加密函数将账号进行加密，将加密后的付款账号在 Bill 表中进行检索，将匹配的信息通过 BillInfo 类中 getBillInfo 函数分别进行 AES 解密，将解密后的信息添至 list 中返回控制层 AccountInfoController.java，再通过 request.setAttribute()发送至 view 层 query_billinfo.jsp 中显示给用户。

主要代码如下：

```
    public ArrayList<BillInfo> getBillInfo(String code,String pwd) throws
UnsupportedEncodingException { //code 是加密后付款账号，binno 是未加密的收款
账号
            ArrayList<BillInfo> list=new ArrayList<BillInfo>();
            String sql="select * from bill where boutno='"+code+"' order by
```

```
bdate desc";//通过加密后付款账号检索匹配的账单信息
            JDBCBean jdbc=new JDBCBean();
            ResultSet rs=jdbc.executeQuery(sql);//进行数据库查询
            try {
                while(rs.next()){
                    BillInfo bi=new BillInfo();
                    bi.setBno(rs.getInt("bno"));//未加密部分直接读取
                    bi.setBoutno(bi.decrypt(rs.getString("boutno"),pwd));
                      //加密部分需进行 AES 解密后读出
                    ……
                    list.add(bi);//查询到的账单信息添至 list 中
                }
            }catch (SQLException e) {
                e.printStackTrace();
            }
            jdbc.close();//关闭数据库连接
            return list;//返回查询到的结果
        }
```

查询历史账单界面如图 5-6 所示。

图 5-6　查询历史账单界面

5.6 修改密码模块的编码实现

5.6.1 修改密码模块功能

该模块主要是将输入的旧密码，新密码和确认密码比对，格式符合后进行更新数据库操作，从而修改密码，下次系统登录时即可用新的密码进行登录。

5.6.2 实现方法

用户修改密码时需要填写修改密码表单，其中包括旧的密码，新的密码和确认密码。同样，后端接收到前端提交的信息参数后，会对接收到的信息进行判断，读取 session 中的账号，与旧密码在账户信息表中匹配检索，符合要求即可进行修改。满足新密码和确认密码相同要求后，系统后端对新密码进行 MD5 加密，将此密文信息取代更新账户信息表中配对的密码，更新成功后即修改密码成功。

例如，用户进行修改密码操作时，点击进入修改密码页面 edit_pwd.jsp，填写表单（旧的密码：123456 新的密码:112233 确认密码：112233），检测表单信息不为空的前提下，检测新的密码和确认密码一致后，提交信息，传递前端页面参数至后端，控制层 AccountInfoController.java 接收到 edit_pwd.action，通过 request.getSession().getAttribute("ano")读取系统登录账号，request.getParameter 读取填写的信息，创建一个 AccountInfo 对象，调用 AccountInfo 类中 MD5_encrypt 函数对旧密码进行加密，将通过此账号在 Account 表中检索的密码与之比较，不一致则提示账户原密码错误，反之，开始进行修改密码操作，调用 AccountInfo 类中 MD5_encrypt 函数对新密码进行加密，通过 AccountInfo 类中 editpwd()函数对 Account 表中密码字段进行更新修改，控制层根据返回的参数判断修改是否成功，成功则跳转至 view 层成功界面 editpwd_success.html，反之，跳转至失败界面 editpwd_fail.jsp。

主要代码如下：

```
 public int editpwd(){
              String sql="update account set apwd='"+apwd+"' where
ano='"+ano+"' ";//设置更新密码的 SQL 语句
              JDBCBean jdbc=new JDBCBean();
              int result=jdbc.executeUpdate(sql);//连接数据库进行更新操作
              jdbc.close();//关闭数据库连接
              return result;//返回修改结果
              }
```

修改密码界面如图 5-7 所示。

图 5-7　修改密码界面

5.7 加解密算法的编码实现

该部分主要介绍 MD5，AES 加密和解密的代码实现。

5.7.1 MD5 加密算法编码实现

MD5 加密算法内部实现流程在此不详细论述了，输入字符串进行加密，加密后得到的字节数组需要转化为 String 类型进行存储。[12]

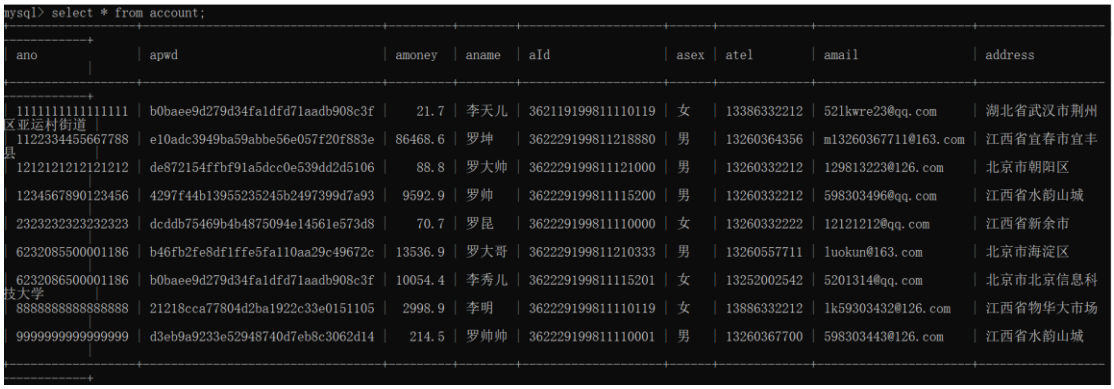
主要代码如图 5-8 所示：

```
//用于账户密码MD5加密
public static String MD5_encrypt(String content) throws Exception {
    MessageDigest md5=MessageDigest.getInstance("MD5");
    byte[] srcBytes=content.getBytes("utf-8");
    byte[] resultBytes = md5.digest(srcBytes);

    //字节数组转String时密文字母小写
    StringBuilder hex = new StringBuilder(resultBytes.length * 2);
    for (byte b : resultBytes) {
        if ((b & 0xFF) < 0x10) hex.append("0");
        hex.append(Integer.toHexString(b & 0xFF));
    }
    return hex.toString();
```

图 5-8　MD5 加密算法实现

数据库账户信息表密码经过 MD5 加密存储后如图 5-9 所示。



mysql> select * from account;

ano	apwd	amoney	aname	aId	asex	atel	amail	address
1111111111111111	b0baee9d279d34fa1dfd71aadb908c3f	21.7	李天儿	362119199811110119	女	13386332212	521kwre23@qq.com	湖北省武汉市荆州区亚运村街道
1122334455667788	e10adc3949ba59abbe56e057f20f883e	86468.6	罗坤	362229199811218880	男	13260364356	m13260367711@163.com	江西省宜春市宜丰县
1212121212121212	de872154ffbf91a5dcc0e539dd2d5106	88.8	罗大帅	362229199811121000	男	13260332212	129813223@126.com	北京市朝阳区
1234567890123456	4297f44b13955235245b2497399d7a93	9592.9	罗帅	362229199811115200	男	13260332212	598303496@qq.com	江西省水前山城
2323232323232323	dcddb75469b4b4875094e14561e573d8	70.7	罗昆	362229199811110000	女	13260332222	12121212@qq.com	江西省新余市
6232085500001186	b46fb2fe8df1ffe5fa110aa29c49672c	13536.9	罗大哥	362229199811210333	男	13260557711	luokun@163.com	北京市海淀区
6232086500001186	b0baee9d279d34fa1dfd71aadb908c3f	10054.4	李秀儿	362229199811115201	女	13252002542	5201314@qq.com	北京市北京信息科技大学
8888888888888888	21218cca77804d2ba1922c33e0151105	2998.9	李明	362229199811110119	女	13886332212	1k59303432@126.com	江西省物华大市场
9999999999999999	d3eb9a9233e52948740d7eb8c3062d14	214.5	罗帅帅	362229199811110001	男	13260367700	598303443@126.com	江西省水前山城

图 5-9　MD5 加密效果图

5.7.2 AES 加密算法编码实现

AES 加密算法内部实现流程在此不详细论述了，输入参数包括待加密信息和密钥种子，本系统将账户密码作为账单信息加密密钥的种子，通过这个种子产生密钥，AES 加密后得到的字节数组需要转化为 String 类型进行存储。[13]

主要代码如图 5-10 所示：

```
//AES加密
//输入：content是需要加密的字符串,password是根据用户密码产生的密钥；输出：加密后的十六进制字符串
public String encrypt(String content, String password) { //content是需要加密的字符串,password是根据用户密码产生的密钥
    try {
        KeyGenerator kgen = KeyGenerator.getInstance("AES");// 创建AES的Key生产者
        kgen.init(128, new SecureRandom(password.getBytes()));// 利用用户密码作为随机数初始化出128位的key生产者
        //加密没关系，SecureRandom是生成安全随机数序列，password.getBytes()是种子，只要种子相同，序列就一样，所以解密只要有password就行
        SecretKey secretKey = kgen.generateKey();// 根据用户密码，生成一个密钥
        byte[] enCodeFormat = secretKey.getEncoded();// 返回基本编码格式的密钥，如果此密钥不支持编码，则返回null
        SecretKeySpec key = new SecretKeySpec(enCodeFormat, "AES");// 转换为AES专用密钥
        Cipher cipher = Cipher.getInstance("AES");// 创建密码器
        byte[] byteContent = content.getBytes("utf-8");
        cipher.init(Cipher.ENCRYPT_MODE, key);// 初始化为加密模式的密码器
        byte[] result = cipher.doFinal(byteContent);// 加密

        //将字节矩阵转换成string类型(将二进制转换成十六进制)
        StringBuffer sb = new StringBuffer();
        for (int i = 0; i < result.length; i++) {
            String hex = Integer.toHexString(result[i] & 0xFF);
            if (hex.length() == 1) {
                hex = '0' + hex;
            }
            sb.append(hex.toUpperCase());
        }
        return sb.toString();
```

图 5-10　AES 加密算法实现

数据库账单信息表转账信息经过 AES 加密存储后如图 5-11 所示。

图 5-11　AES 加密效果图

5.7.3 AES 解密算法编码实现

AES 解密过程即加密的逆过程，前文也对 AES 解密过程详细论述过，这里就不详细论述了。[14]

主要代码如图 5-12 所示：

```
//AES解密
//输入：code是需要解密的十六进制字符串,password是根据用户密码产生的密钥；输出：明文
public String decrypt(String code, String password) throws UnsupportedEncodingException {
    try {
        //将string类型转换成字节矩阵(将十六进制转换成二进制)
        if (code.length() < 1)
            return null;
        byte[] content = new byte[code.length() / 2];
        for (int i = 0; i < code.length() / 2; i++) {
            int high = Integer.parseInt(code.substring(i * 2, i * 2 + 1), 16);
            int low = Integer.parseInt(code.substring(i * 2 + 1, i * 2 + 2), 16);
            content[i] = (byte) (high * 16 + low);
        }

        KeyGenerator kgen = KeyGenerator.getInstance("AES");// 创建AES的Key生产者
        kgen.init(128, new SecureRandom(password.getBytes()));
        SecretKey secretKey = kgen.generateKey();// 根据用户密码，生成一个密钥
        byte[] enCodeFormat = secretKey.getEncoded();// 返回基本编码格式的密钥
        SecretKeySpec key = new SecretKeySpec(enCodeFormat, "AES");// 转换为AES专用密钥
        Cipher cipher = Cipher.getInstance("AES");// 创建密码器
        cipher.init(Cipher.DECRYPT_MODE, key);// 初始化为解密模式的密码器
        byte[] result = cipher.doFinal(content);
        String de_code = new String(result,"UTF-8");
        return  de_code;// 明文

    } catch (NoSuchAlgorithmException e) {
        e.printStackTrace();
```

图 5-12　AES 解密算法实现

第六章　系统测试

本章主要内容为对系统进行软件测试，查缺补漏。

测试是发现并指出系统缺陷和错误的过程，根据系统开发过程中各阶段的需求文档和结构说明设计一些测试用例，并通过这些测试用例找出系统错误，测试是软件、系统开发过程中不可缺少的关键一环。

软件测试主要分为静态分析和动态测试。其中，静态分析主要是对程序代码的检查，有审查，走查和评审等形式。动态测试则有白盒测试和黑盒测试这两种形式。

白盒测试，是针对程序内部逻辑和结构进行更加细致的检测，一般用于单元测试，其中，基本路径测试使其覆盖每条逻辑分支可能，从而细致地找出每个模块的错误。

黑盒测试，则是不考虑程序内部的逻辑结构，将程序看做一整个黑盒，只针对于系统的输入和输出是否符合系统预期的要求，一般是功能性测试。

系统测试则需要将白盒测试和黑盒测试结合起来，一起完成系统测试。

6.1 系统模块测试

6.1.1 注册模块测试

注册需要填写相关账户相关信息，包括账号，密码，确认密码，用户姓名，ID，性别，电话，邮箱，地址。针对以上信息，设计出一系列测试用例，格式出现错时提示错误信息，具体如表 6-1 所示。

表 6-1　注册模块测试情况表

测试说明 测试用例 ID：001　　操作系统：Windows 10 子系统：注册字符　　测试日期：2020-5-18
初始设置 1. 点击注册按钮,进入注册界面。
输入信息 1. 账号：9087556421333330　密码：lklk11　确认密码：lklk11　姓名：罗缇　ID：123456789012345678　性别：男　电话：13423121177　邮箱：12343@qq.com　地址：新昌街 33 号　　提交 2. 账号：　密码：lklk11　确认密码：lklk11　姓名：罗缇　ID：123456789012345678　性别：男　电话：13423121177　邮箱：12343@qq.com　地址：新昌街 33 号　　提交 3. 账号：9087556421aaaaa1　密码：lklk11　确认密码：lklk11　姓名：罗缇　ID：123456789012345678　性别：男　电话：13423121177　邮箱：12343@qq.com　地址：新昌街 33 号　　提交 4. 账号：9087556421333332　密码：　确认密码：　姓名：罗缇　ID：123456789012345678　性别：男　电话：13423121177　邮箱：12343@qq.com　地址：新昌街 33 号　　提交

5. 账号：9087556421333333 密码：lklk11 确认密码：lklk22 姓名：罗缇 ID：123456789012345678 性别：男 电话：13423121177 邮箱：12343@qq.com 地址：新昌街 33 号 提交 6. 账号：9087556421333334 密码：lklk11 确认密码：lklk11 姓名： ID：123456789012345678 性别：男 电话：13423121177 邮箱：12343@qq.com 地址：新昌街 33 号 提交 7. 账号：9087556421333335 密码：lklk11 确认密码：lklk11 姓名：罗缇 ID： 性别：男 电话：13423121177 邮箱：12343@qq.com 地址：新昌街 33 号 提交 8. 账号：9087556421333333 密码：lklk11 确认密码：lklk11 姓名：罗缇 ID：123456789012345678 性别：男 电话： 邮箱：12343@qq.com 地址：新昌街 33 号 提交 9. 账号：9087556421333333 密码：lklk11 确认密码：lklk11 姓名：罗缇 ID：123456789012345678 性别：男 电话：13423121177 邮箱： 地址：新昌街 33 号 提交 10. 账号：9087556421333333 密码：lklk11 确认密码：lklk11 姓名：罗缇 ID：123456789012345678 性别：男 电话：13423121177 邮箱：12343@qq.com 地址： 提交
预期结果 1. 注册成功 2. 提示“账号不能为空” 3. 提示“格式错误” 4. 提示“密码不能为空” 5. 提示“密码不一致” 6. 提示“姓名不能为空” 7. 提示“身份证号码不能为空” 8. 提示“电话不能为空” 9. 提示“邮箱不能为空” 10. 提示“地址不能为空”
实际结果 通过（10） 失败（0）

6.1.2 登录模块测试

用户在进行登录操作时，需要输入账号和密码。针对以上信息，设计一系列测试用例，如表 6-2 所示。

表 6-2 登录模块测试情况表

测试说明	
测试用例 ID：002	操作系统：Windows 10
子系统：登录字符	测试日期：2020-5-18
初始设置	

1. 进入系统初始登录页面。
输入信息 1. 账号：1122334455667788　密码：123456　　登录 2. 账号：1122334455667788　密码：222222　　登录 3. 账号：1122334455667788　密码：　　登录 4. 账号：1122334455aaaaa8　密码：123456　　登录
预期结果 1. 登录成功，进入系统主页 2. 登陆失败，返回登录界面 3. 登陆失败，返回登录界面 4. 登陆失败，返回登录界面
实际结果　　通过（4）　　失败（0）

6.1.3 转账模块测试

用户在进行转账操作时，需要输入收款账号，收款人姓名，转账金额和备注。针对以上信息，设计一系列测试用例，如表 6-3 所示。

表 6-3　转账模块测试情况表

测试说明 测试用例 ID：003　　操作系统：Windows 10 子系统：转账字符　　测试日期：2020-5-18
初始设置 1. 登录系统成功进入系统主页。 2. 选择转账业务，填写账单信息。
输入信息 1. 收款账号：1234567890123456　收款人姓名：罗帅　转账金额：33　备注：你好啊兄弟　　提交 2. 收款账号：1234567890123456　收款人姓名：罗帅　转账金额：-33　备注：你好啊兄弟　　提交 3. 收款账号：1234567890123456　收款人姓名：罗帅　转账金额：aa　备注：你好啊兄弟　　提交 4. 收款账号：1234567890123456　收款人姓名：罗天儿　转账金额：33　备注：你好啊兄弟　　提交 5. 收款账号：　收款人姓名：　转账金额：　备注：你好啊兄弟　　提交

预期结果 1. 转账成功，5 秒后自动返回系统主页 2. 转账失败，提示“转账金额不能为负数，请重新输入” 3. 转账失败 4. 转账失败，提示“收款账号与收款人姓名不符，请重新输入” 5. 转账失败		
实际结果	通过（5）	失败（0）

6.2 加密算法安全测试

对于本系统的信息安全存储问题，通过加密算法进行加密，主要是账户密码的 MD5 加密，转账信息的 AES 加密。对于加密算法的实现与应用，通过对信息加密之前的明文和加密之后的密文对比测试，以及信息加密之前和解密之后对比测试，找出可能存在的问题，对比加密前后的信息保密程度。

MD5 加密账户密码之前和加密之后信息测试对比如图 6-1 所示，AES 加密之前明文信息与加密之后密文信息测试对比如图 6-2 所示，AES 加密之前信息与解密之后信息测试对比如图 6-3 所示。

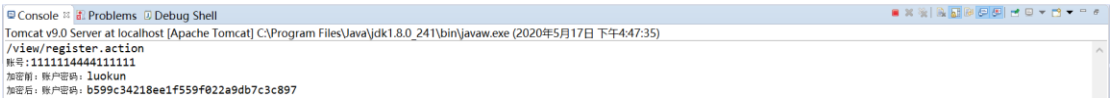


图 6-1 MD5 加密前后对比图

图 6-2 AES 加密前后对比图

图 6-3 AES 加解密对比图

由此可见，本系统的加密和解密程序能够成功运行，对于加密之前和解密之后的信息能够保证信息的一致性，不会造成因加解密问题导致账单信息不符。同时，对比加密之前和加密之后的信息，可以明显发现加密之后的信息的保密性极高，差别极大，成功起到了信息保密存储的预期需求。

第七章　总结

7.1 课题总结

本课题针对信息安全存储等安全问题展开研究，加强线上转账业务的安全和信息保护，能够在交易之余不会出现信息泄露,信息贩卖等问题,设计一个安全的模拟转账系统。最后，本课题完成情况如下：研究了转账服务的类型，过程及安全方法，针对信息安全存储这一安全研究方面，对相关加密算法解密算法的研究与应用实现，研究分析了 AES，RSA，MD5 加密算法的流程及每一步，每一次循环的机制原理，和调查分析了电子货币转账的技术过程，设计并实现了一个基于 AES，MD5 加密算法的小型模拟电子货币转账的系统。在完成通过一种 AES 加密算法加密信息的转账系统之余，研究分析了其他种类的加密算法，并采用了 MD5 进行身份认证，与 AES 结合使用，提高信息加密能力，进而保障系统信息安全。

在整个系统开发过程中，我学到了许多知识，掌握了 MVC 设计模式，对前后端分离开发进一步的了解，熟料使用和掌握 MySQL，eclipse 等开发工具，掌握相关加密算法的原理流程，对 web 项目开发有了进一步的深入。Jsp+MVC+MySQL 这一开发设计模式比较适合开发小型项目，通过对此的学习和研究，深刻明白了前端和后端的工作原理和工作模式。

这一系统整个开发过程离不开软件工程的知识，从问题定义，可行性研究，需求分析到软件设计和编码，最后通过软件测试完善整个系统，可见软件工程贯穿整个系统开发，指导系统在不偏离主线的基础上逐步完善，从而设计出与预期相符的系统。

7.2 系统不足及改进

由于今年疫情情况十分严峻，导致整个毕设开发过程都是在家通过线上方式完成的，使得整个系统开发过程中遇到问题时，更加需要自己查阅网上相关资料加以解决，同时老师也对我帮助颇多，而在系统开发的同时，又需要准备研究生复试工作，多重压力导致系统没有尽善尽美，十分抱歉。

系统的主要不足与改进措施如下：

1、该系统主要面向用户这一角度进行开发，可以设计管理员身份进行登录管理，使得能有相关人员有权限访问和管理系统。

2、对于几种加密算法的性能和表现，可以不仅通过加密前后进行对比，还可以进行详细地量化分析，通过 MATLAB 来进一步对比各加密算法的参数情况。

3、系统界面设计仍需改进，整体框架划分也需要改进，可以参考一些好的系统布局，尽可能多地了解每个控件的使用。

4、通过 MVC 模式实现了前后端分离，但是 JSP 页面中 js 编程使得前端页面仍然残余部分后端的编程代码，没有彻底地进行分离，可以通过更新技术方法，采用更加前沿的技术来满足这一结构化的开发需求。

致谢

随着毕业设计的完成，大学四年的本科学习也就告一段落了。回首过去，从大一的懵懵懂懂到如今即将步入社会或者继续研究生阶段的深造学习，从中学习到了很多专业知识和人生道理，经过四年的磨练，使我在各个方面得到极大的锻炼。在此，我特别要感谢在大学期间给予我帮助和鼓励的人们。

首先，我特别要感谢我的毕设指导老师牛欣源。从最初的毕设选导师，选研究课题到后来的系统构建，以及完成任务的过程中，牛欣源老师不遗余力地对我进行指导教学。虽然因为疫情导致老师只能和我们线上沟通，但是牛欣源老师依然能够在百忙之中准时开设毕设相关线上会议，对我们的毕设开发过程进行指导督促。对于毕设任务开展之余，因考研复试工作耽误一些毕设进程，牛老师十分理解并支持鼓励我，所以，我特别感谢牛欣源老师。

其次，我也十分感谢大学四年期间的老师和同学。在整个大学学习生活中，老师的教导和同学的帮助，使得我在大学期间能够学习到很多专业相关知识，以及在面对各种压力时能够及时缓解，积极主动地投入到新的学习生活当中。

再次，我特别感谢我的家人，我的爸爸，妈妈和姐姐。在大学四年期间，不管我面对什么压力，他们总是站在我这边，理解并支持我的学习工作，使得我在学习时没有顾虑，让我在能够取得不错的成绩。今后，我一定会更加努力。

最后，我也十分感谢我的评阅老师和答辩老师能在百忙之中抽空评审我的毕业设计和毕业论文，毕业论文中存在的不足之处，还请各位老师能够批评指正，由衷地感谢！

参考文献

[1] Nicholas C.Zakas. JavaScript 高级程序设计[M],北京:人民邮电出版社.2006

[2] Jeremy Keith. JavaScriptDOM 编程艺术[M],北京:人民邮电出版社.2007

[3]刘慧.我国商业银行网上银行业务发展对策研究[D].2011:3-30

[4]郝璐军.基于 AES 的物流管理系统设计与实现[D].2008:2-70

[5]赵雪梅.AES 加密算法的实现及应用[J].2010，24（2）：105-110

[6] 费希利（美）.SQL 基础教程[M],北京：人民邮电出版社,2009

[7]孙卫琴. Tomcat 与 JavaWeb 开发技术详解[M],电子工业出版社，2009

[8]范丽辰,王桂海.AES 密钥扩展算法研究与改进[D],山东，2015：51-55

[9]高娜娜,宋丽华.AES 算法中密钥扩展和列混合单元的可重构设计[J].北京信息科技大学学报.2012，27（4）：51-55

[10]孙爱娟. 基于 AES 加密算法的改进及其 MATLAB 实现[D].哈尔滨理工大学.2009:30-70

[11]孟敬. 基于 MD5 算法的自动取款机安全系统的设计与实现[J].长沙大学学报.2012，26（5）：60-62

[12]郭克华,刘小翠,唐雅媛.Java 程序设计与应用开发[M],北京：清华大学出版社.2018

[13] O Baudron.，H Gilbert. Report on the AES Candidates[J]. Second Advanced Encryption Standard Candidate Conference . 1999

[14] Lavanya R,Karpagam M. Enhancing the security of AES through Small Scale Confusion Operations for Data Communication[J]. Microprocessors and Microsystems.2020