

# 1 引言

## 1.1 编写目的

从本阶段开始，项目进入正式开发阶段。这份总体设计报告的编写目的，是以本项目的需求分析说明书为依据，从总体设计的角度，明确系统管理子系统的总体架构、流程、数据结构、数据库设计。

目的在于:

- 为开发人员提供依据
- 为修改、测试、维护提供条件
- 明确各模块外部接口，内部接口，用户接口
- 项目负责人按计划说明书的要求布置和控制开发工作全过程

本说明书的预期读者包括:

- 软件客户
- 项目经理
- 测试人员
- 项目管理和开发人员
- 系统维护人员

## 1.2 项目背景

软件系统名称

- 股票交易管理系统

任务提出者

- 软件工程基础课程任课老师——王新宇

软件开发者的

- 浙江大学2021-2022学年夏学期软件工程基础课程学生C1项目组

实现该软件的计算机网络

- 由若干台PC机组成的局域网

## 项目相关背景介绍

浙江大学软件工程基础课程分为理论课与实践课两个部分。在理论课中，教师有选择地介绍了与软件工程基础相关的理论;强调并确定了适用于整个软件生命期的基本原则，全面而深入地介绍了这些基本原则在软件设计、规范、验证、软件生产过程和管理活动中的运用。而实验课采取分组形式完成，每5个学生为一组，分别设有组长、主程序员、程序员、测试员、文档员等角色。本次课程，教师选取股票交易管理系统作为综合性实验题目。其中我组主要负责证券开户、资金开户两个子系统，实现对于用户个人信息的数据库管理操作。

## 1.3 相关定义

### (1) MySQL

一个小型关系型数据库管理系统。

### (2) JavaScript

JavaScript是一种面向对象的动态类型的区分大小写的客户端脚本语言。

### (3) SQL注入

通过把SQL命令插入到Web表单递交或输入域名或页面请求的查询字符串，最终达到欺骗服务器执行恶意的SQL命令。

### (4) 数据库连接池

数据库连接池负责分配、管理和释放数据库连接，它允许应用程序重复使用一个现有的数据库连接，这项技术能明显提高对数据库操作的性能。

### (5) UTF-8

UTF-8是UNICODE的一种变长字符编码，又称万国码安全证书。安全证书是在进行网上交易时的身份证，或者说是私人钥匙，安全证书是唯一的，与任何其他人的证书都不相同。

### (6) MD5加密算法 Message Digest Algorithm

MD5(消息摘要算法第五版)为计算机安全领域广泛使用的一种散列函数，用以提供消息的完整性保护。

### (7) React

React是用于构建用户界面的JavaScript库，起源于Facebook的内部项目，该公司对市场所有JavaScript MVC框架都不满意，决定自行开发一套，用于假设Instagram的网站。于

2013年5月开源。由于React的设计思想极其独特，属于革命性创新，性能出众，代码逻辑却非常简单。所以，越来越多的人开始关注和使用，认为它可能是将来Web开发的主流工具。

#### (8) Express

Express 是一个简洁而灵活的node.js Web应用框架，提供一系列强大特性帮助你创建各种Web应用。Express 不对 node.js 已有的特性进行二次抽象，还在它之上扩展了Web应用所需的功能。丰富的HTTP工具以及来自Connect框架的中间件随取随用，创建强健、友好的API变得快速又简单。

## 1.4 系统概述

股票交易管理系统基于完整经济模型，为各地股民提供线上交易服务，具有买卖交易股票、信息交流等功能。系统需要持续、稳定地运行，延时性小，否则将无法满 足用户股票磋商交易。系统一共由六个模块构成，分别为证券账户开户系统、资金账户开户系统、交易客户端、股票中央交易系统、网上信息发布系统、交易系统管理。

#### (1) 证券账户开户系统

办理证券帐户是走进股市的第一步。证券帐户可以视为投资者进入股票交易市场的通行证，只有拥有证券账户，才能进场买卖证券。想要开设证券帐户，开户人必须不是国家规定禁止开户的四类人员，其他人员均可以凭 相关证件到交易所申请开设证券帐户。证券帐户分为两种，分别是自然人开设的个人帐户，以及法人帐户。每个投资者，都只能开立一个证券帐户，并仅限于本人使用。自然人开立的证券帐户为个人帐户。开立个人帐户时，投资者必须持有效的身份证件(一般为居民身份证)去证券交易所指定的证券登记机构或会员证券公司办理名册登记并开立证券帐户。

#### (2) 资金账户开户系统

投资者委托买卖股票，须事先在证券经纪商处开立证券交易结算资金帐户，资金帐户用于投资者证券交易的资金清算，记录资金的币种，余额和变动情况。资金账户开设时，需提交本人身份证与证券账户卡，并将资金帐户和证券帐户相关联，投资者在资金帐户中的存款可随时提取，证券经纪商按活期存款利率定期计付利息并自动转入投资者的资金帐户。投资者委托买入时，资金帐户要有足够的余额。资金帐户需要设置交易密码，以便在交易客户端使用。另外，还需要设置及取款密码，用于从资金帐户里面取出 资金。帐户开设完毕，用户会得到一张类似于银行卡的，资金帐户卡片。通过在证券经纪商 指定的交易客户端设备上 面刷卡，并输入资金帐户设置的密码登陆，便可以发出买卖股票的 指令，查看资金帐户的资金情况和证券帐户内的股票信息。

资金帐户密码可以有两张方法修改。1. 投资者可以在证券经纪商处在开设资金帐户的设备上，通过输入原有密码和新密码来修改。2. 投资者还可以在证券交易客户端通过输入原有密码和新密码修改。

#### (3) 交易客户端

投资者在开办完证券帐户并在证券经纪商处开立证券交易结算资金帐户以后，就可以通过交易客户端委托证券经纪商进行股票的买卖交易了。目前存在的委托方式有很多种，主要有以下3种：

- 交易设备：投资者持资金帐户卡在证券经纪商指定的刷卡设备处刷卡，并输入验证密码。然后就可以发出委托，进行交易。这是最早，也是最常见的委托方式。

- 网络：投资者通过登陆证券经纪商指定的网址，通过必要的用户和密码验证登陆。然后发出可以发出委托，进行交易。

- 电话：投资者通过申请，然后就可以致电证券经纪商指定的电话号码，通过必要的用户和密码验证，然后根据语音提示，发出委托。

用户登陆客户端之后，可以查询证券账户持有的股票。界面将会显示股票的名称、总数，现在价格、持有成本与持有股票损益。用户还可以查看资金账户内的资金信息，并且根据资金与股票信息发出购买、出售股票的指令进行交易。

#### (4) 股票中央交易系统

中央交易是整个股票交易过程的核心部分。所有投资者发出的买卖股票指令都在这个自动撮合系统参加集合竞价或连续竞价，交易系统根据时间优先及价格优先的原则，对符合条件的指令予以成交，这个过程就是撮合成交。股票成交后，中央交易系统随后将成交记录反馈到相关的股票帐户和证券经纪商的资金帐户。股票中央交易系统还要设置涨跌停保护机制，保证股票交易市场的相对稳定性，所以每天股票的最高交易价格与最低交易价格不能超过该限制范围。

#### (5) 网上信息发布系统

为了将交易信息能够更好地提供给各种研究机构和广大股民，需要通过建立网站实现交易信息网上发布系统。并且通过对不同用户的认证，提供不同类型的交易和统计信息。用户可以通过登录网上信息发布系统，查询股票实时的价格。股票信息通常有代码信息、K线图等等，为用户交易提供信息来源。

#### (6) 交易系统管理

交易系统管理模块实现的是对交易系统进行监控的功能，它是一个内部管理模块，用户是股票交易所的内部人士。由于需要在适当的时候通过调整买卖双方的价格比例来促进交易，所以需要提供一个接口给特别的授权用户来查看不同股票的实时信息。

## 1.5 项目概述

本小组负责股票交易管理系统当中的证券账户开户系统与资金账户开户系统。证券开户系统与资金开户系统将面对管理员开放，管理员可以根据真实的用户信息对账户进行开户操作，并分配账号供用户登陆。

证券开户系统与资金账户开户系统是整个交易管理系统中一个较为底层的模块，将负责管理用户个人信息的数据库，同时也要为其他系统提供查询登录的借口。由于数据库信息的私密性，证券账户开户与资金账户开户系统仅仅支持管理员登录，默认假设用户是在证券市场现场进行开户操作的。

## 2 总体设计

### 2.1 需求规定

#### 2.1.1 系统功能

根据需求，本项目需要提供维护证券账户信息的数据库、维护资金账户信息的数据库，完成网页的前后端呈现。API方面，本项目主要需要提供用户登录证券账户证券账户、登录资金账户的接口，交易时查询、修改资金账户余额的接口。前段界面主要设计成管理员登录界面、证券账户开户界面、资金账户开户界面，完成两个子系统的的功能。

#### 2.1.2 系统性能

界面设计应简洁直观，布局合理，清晰地呈现信息，突出重点内容。操作方便，用户容易上手。系统具有良好的反应速度，给用户良好的使用体验。我们要求在网络情况下，系统应具有以下时间特性要求：

(1) 单个用户在线时Web响应用户动作时间小于1秒。数据发布操作响应用户动作时间小于1秒。

(2) 500个用户同时在线时 Web 响应用户动作时间小于2秒。数据发布操作响应用户动作时间小于5秒。

- 访问容量：该系统至少在同一时间内支持 500 个用户并发访问。
- 服务器配置最低要求：CPU2.6G，内存 2.0G，硬盘 7200 转。
- 数据处理能力：至少支持 10000 条数据发布记录。
- 可用性：该子系统应实现在大多数流行的 Web 浏览器中正确显示和执行，包括 Firefox、Chrome、Edge、IE 等。

#### 2.1.3 输入输出要求

客户端通过网页展现给用户一个友好的界面，用户可以通过提交表单或者点击超链接向服务器提供数据与命令。服务器后台处理后将结果显示到用户的网页界面上。API则为其他子系统和前端提供清晰、简洁的接口，子系统通过API向服务器发送请求，服务器后台处

理后返回格式化的结果;若子系统进行非法操作，服务器能够进行判断并返回错误信息，避免发送的请求影响后端稳定性。

## 2.1.4 数据管理能力要求

### 2.1.4.1 安全

#### 保密性

1. 用于身份验证的用户名和密码应防止未经授权的用户访问系统。应构建访问控制以防止合法用户非法使用系统资源。
2. 某些敏感数据(如用户名和密码)在交换时应加密存储。密码的存储也需加密。
3. 在用户登录期间，应该防止 SQL 注入，密码强制破解和伪造会话入侵。

#### 完整性

1. 防止非法用户对数据进行无意或恶意的修改、插入、删除，防止数据丢失。
2. 防止内部用户对数据进行无意或恶意的修改、插入、删除，防止数据丢失。
3. 为数据库加上一定的约束，对关键性操作如删除、修改进行限制，并对用户进行警示。
4. 定期备份数据。

### 2.1.4.2 性能

对于频繁访问数据库的操作，后台需要建立持久的数据库连接，以避免重复连接数据库耗费资源。

## 2.2 运行环境规定

### 2.2.1 服务器端

由于实验条件有限，我们并不能提供专门的服务器运行系统，故将利用配置较高的PC作为服务器，保证服务器以及客户端间网络畅通即可。

#### 计算机

➤ CPU: 不小于2.0GHz

➤ 内存: 不小于2.0GB

#### 软件依赖

- 操作系统: Windows Vista/7/8/8.1/10, Mac OS, Linux
- 数据库平台: MySQL
- 前端开发框架: React
- 后端开发框架: Express
- Web 服务器: Apache
- MySQL 管理软件: MySQL WorkBench等
- 开发工具: 能支持网页开发的工具均可 (如Pycharm、Vscode)
- 测试工具: 能支持测试的工具均可
- 办公软件: Microsoft Office, Typora
- 浏览器: Chrome、Edge、Safari(移动端)

## 2.2.2 客户端

### 外围设备

- 键盘鼠标: 可正常使用
- 显示器: 可正常使用
- 硬盘: 不小于 100GB
- 硬盘转速: 不小于 7200rpm

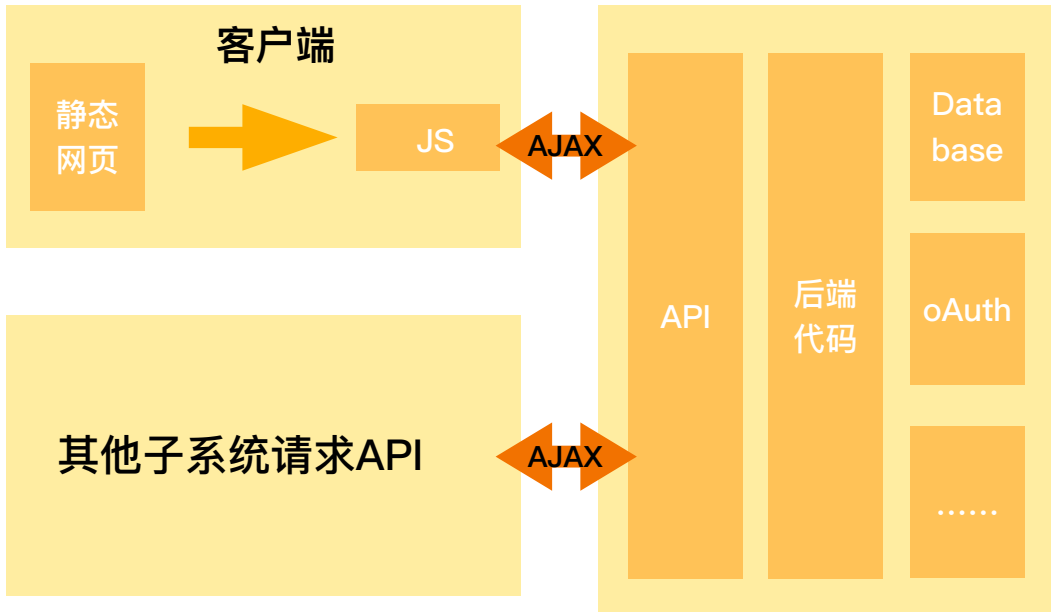
### 通讯设备

- 网线: 正常联通且数据传输能力良好 • 网卡: 100M
- 网卡: 100M

## 2.3 基本设计概念和流程处理

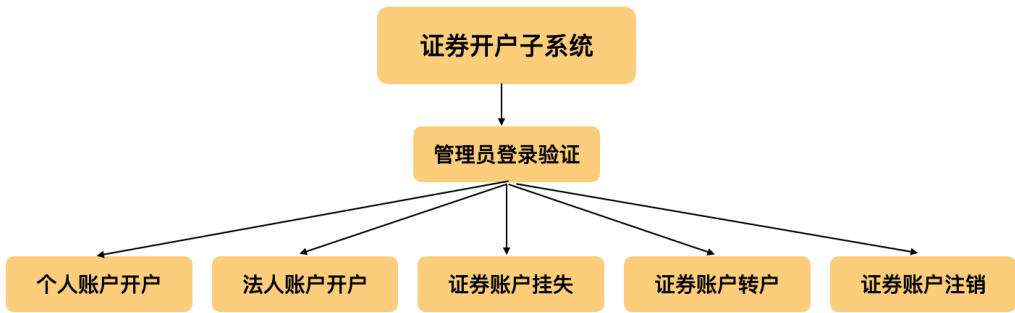
- 本子系统是一个横跨前后端的大模块, 主要负责个人证券账户、公司证券账户、个人资金账户的创建以及数据库管理。
- 前端: React框架
- 后端: 基于Node.js的Express框架
- 数据库: 数据库采用MySQL, 通过promise来实现异步查询

- 客户端:考虑到本系统所使用的框架较新,推荐使用Chrome、Firefox、Safari或Microsoft Edge浏览器。不推荐使用任何一个版本的IE浏览器,如需使用,版本须在11及以上。

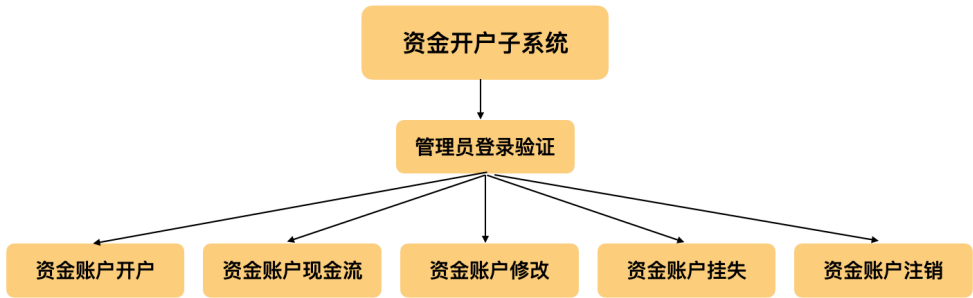


## 2.4 结构

### 2.4.1 证券开户子系统结构图



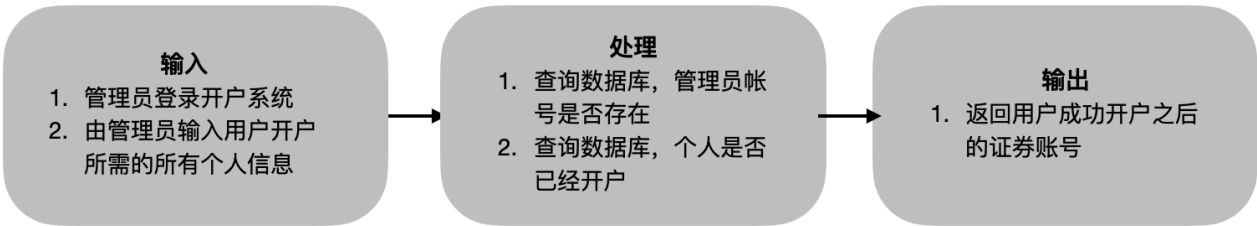
### 2.4.2 资金开户子系统结构图



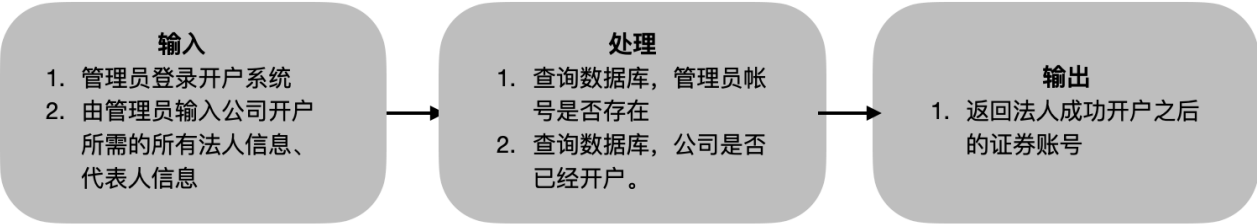


## 2.4.3 功能IPO图

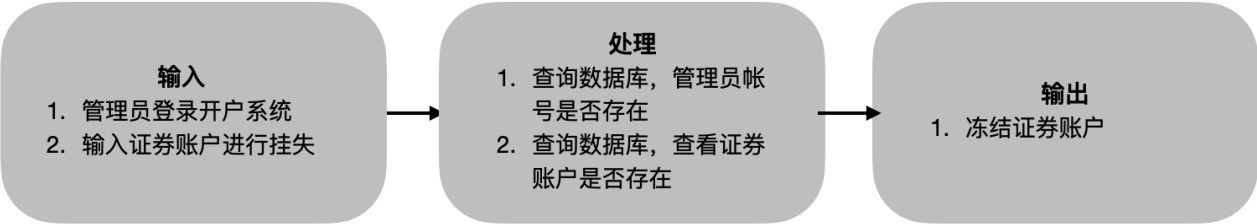
### 2.4.3.1 证券个人账户开户



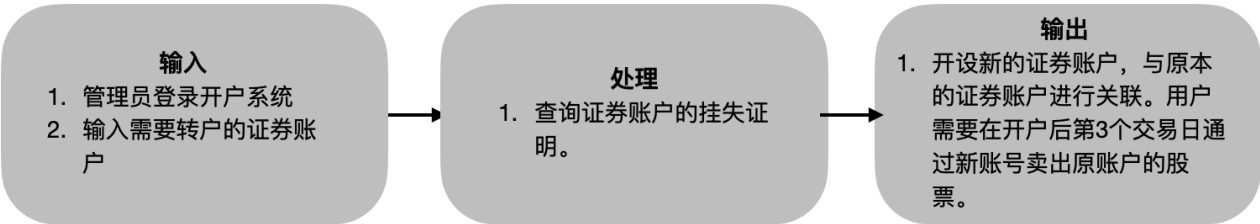
### 2.4.3.2 证券法人账户开户



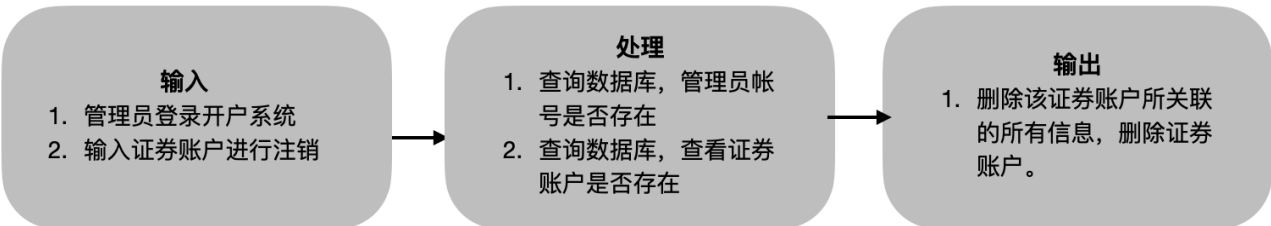
### 2.4.3.3 证券账户挂失



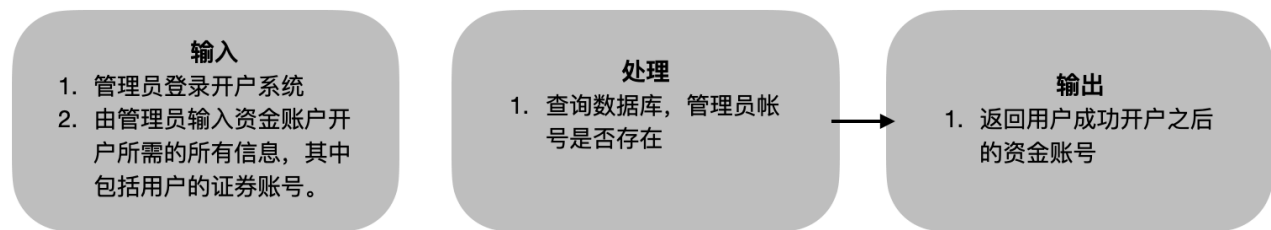
### 2.4.3.4 证券账户转户



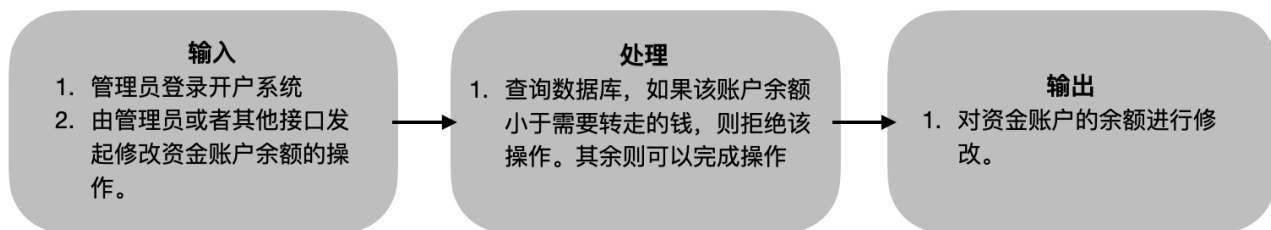
### 2.4.3.5 证券账户注销



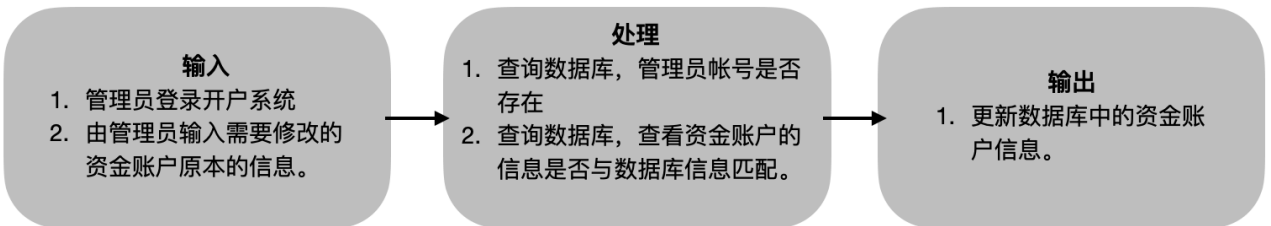
### 2.4.3.6 资金账户开户



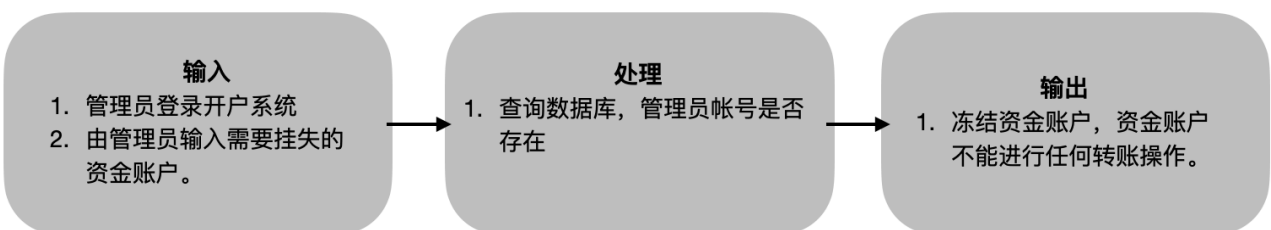
### 2.4.3.7 资金账户现金流处理



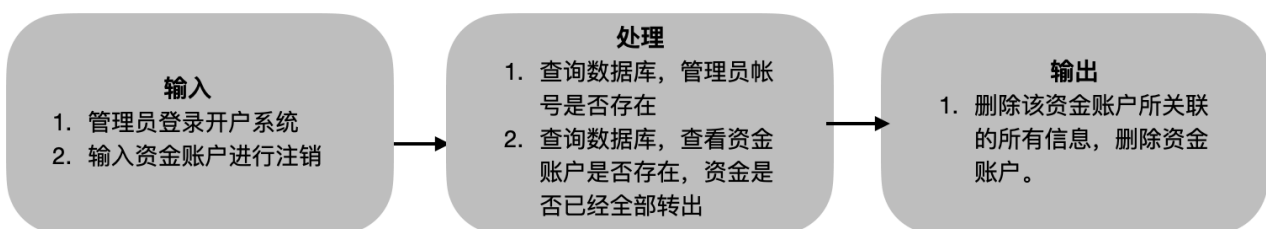
### 2.4.3.8 资金账户修改



### 2.4.3.9 资金账户挂失



### 2.4.3.10 资金账户注销



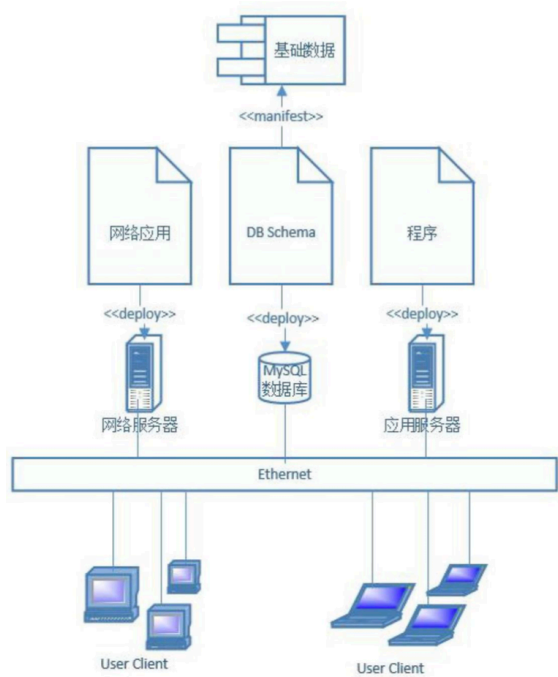


### 3.2 技术简介

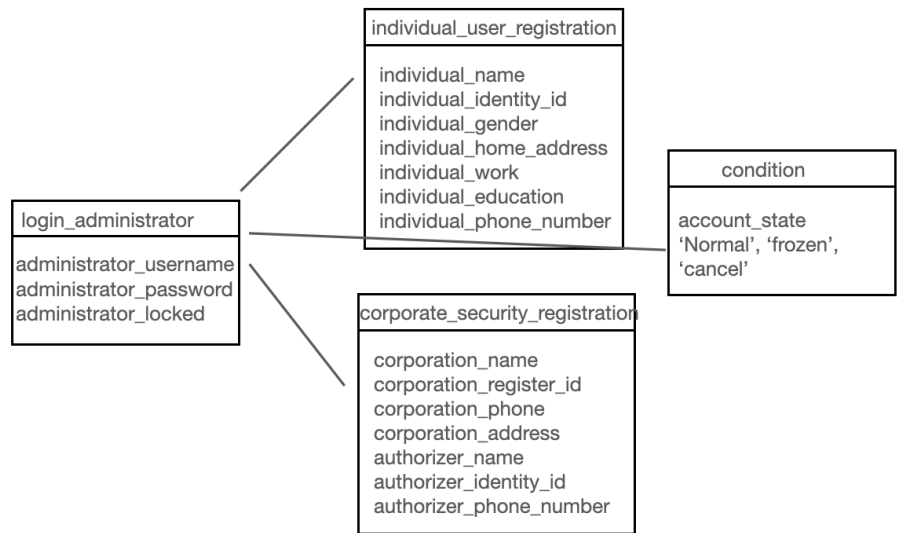
在系统设计中，我们将采用基于 Node.js 的Express框架来进行 Web 页面的构建。Express 是一个开放源代码的 Web 应用框架。使用Express， 只要很少的代码，程序开发人员就可以轻松地完成一个正式网站所需要的大部分内容。它的还带有强大的数据库功能，自带强大的后台功能以及具有优雅的网址。

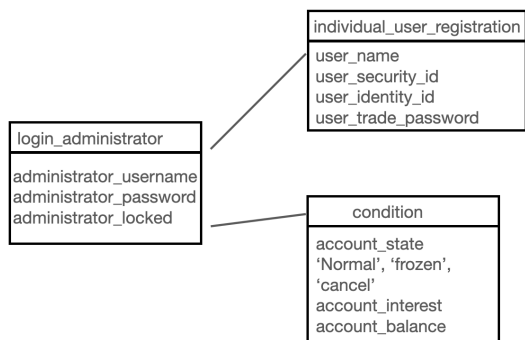
此外，我们将采用模型(Model)，模板(Template)，视图(View)(MTV, 三层架构 模式)模型对整个项目进行设计和实现。MTV模式是软件工程中的一种软件架构模式，把软件系统分为三个基本部分:模型(Model)，模板(Template)，视图(View)。这样一 来，各组件之间就保持了松耦合关系:M主要是用来负责业务对象和数据库对象，T负责把页面如何展示给用户，V负责业务逻辑，并在适当的时候调用M和T。

### 3.3 部署图

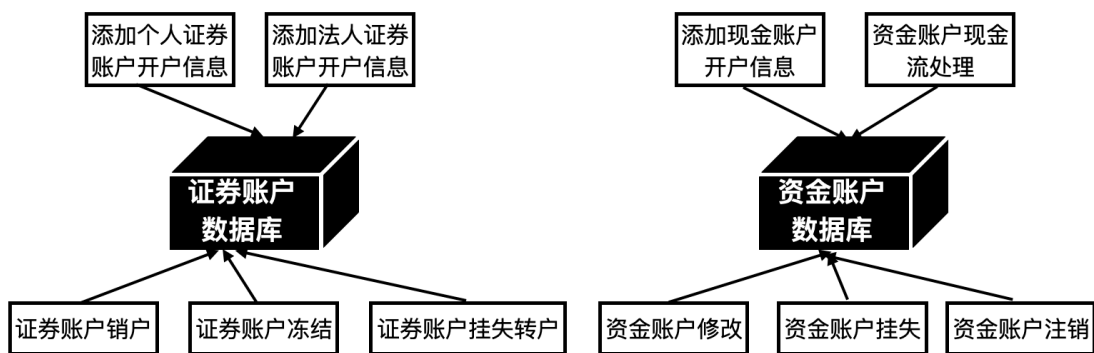


### 3.4 类图





### 3.5 内部接口图



### 3.6 顺序图

