**“xMarket 虚市”**

**详细设计说明书**

|  |  |  |
| --- | --- | --- |
| 文件状态：  [√ ] 草稿  [ ] 正式发布  [ ] 正在修改 | 文件标识： | “虚市”详细设计文档 |
| 当前版本： | 2.0 |
| 作 者： | 郑科宇、戴哲力、张浩、任豪、王佳琪、刘佳 |
| 完成日期： | 2019-05-23 |

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| 1.0 | 郑科宇 | 戴哲力、张浩、任豪、王佳琪、刘佳 | 2019/05/06-2019/05/07 | 仅列出基本框架 |
| 1.1 | 郑科宇 | 戴哲力、张浩、任豪、王佳琪、刘佳 | 2019/05/21-2019/05/23 | 增加了全局通用的代码和错误处理模块。 |
| 2.0 | 郑科宇 | 戴哲力、张浩、任豪 | 2019/05/23-2019/05/23 | 增加了竞价算法。增加了买卖和登录模块的时序图。增加了一些注释，修正了文档中的一些错误。 |
| 2.1 | 郑科宇 | 戴哲力 | 2019/05/23-2019/05/23 | 增加实体类，修改service，repository相关内容。 |

# 引言

## 目的与范围

编写详细设计说明书的目的是为了使开发人员在完成概要设计说明书的基础上，完成各项功能的规定，为了程序员写出实际的程序代码提供依据。它是软件设计阶段所有任务和相关人员所需要的参考资料。预期的读者为系统开发者。

本文档主要包括全局数据结构通用代码说明，实体类的设计，接口，service、repository类相关设计，另外还包括关键模块登录与买卖的时序图，以及整个开发的重点竞价算设计。除此以外，还有一些信息返回格式设计与异常错误模块设计内容。数据库的设计请见《数据库设计文档》。

# 全局数据结构说明

通用代码：

|  |  |  |
| --- | --- | --- |
| Name | Code | Message |
| SUCCESS | 200 | OK |
| UNKNOWN\_ERROR | 10000 | 未知的错误 |
| OBJECT\_ALREADY\_EXIST\_ERROR | 10001 | 要创建的对象已存在 |
| VALIDATION\_ERROR | 10002 | 验证失败 |
| OBJECT\_NOT\_EXIST\_ERROR | 10003 | 要获取的对象不存在 |
| VERIFICATION\_CODE\_FAIL\_ERROR | 50001 | 获取验证码失败 |
| RESET\_PASSWORD\_FAIL\_ERROR | 50002 | 重置密码失败 |
| USERID\_NOT\_EXIST\_ERROR | 50003 | 用户名不存在 |
| EMAIL\_NOT\_EXIST\_ERROR | 50004 | 邮箱不存在 |
| FUND\_ERROR | 50005 | 资金不足 |
| ORDER\_FINISH\_ERROR | 50006 | 委托已完成 |
| ILLEGAL\_TIME\_ERROR | 50007 | 非法的交易时间 |

# 实体类设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | User | 类解释 | 用户信息 | |
| 属性 | | 类型 | | 备注 |
| userId | | int | | 用户id |
| email | | String | | 邮箱 |
| userName | | String | | 用户名 |
| loginPassword | | double | | 登陆密码 |
| transactionPassword | | int | | 交易密码 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | UserFund | 类解释 | 用户资金信息 | |
| 属性 | | 类型 | | 备注 |
| userId | | int | | 用户id |
| balance | | double | | 资金余额 |
| frozenAmount | | double | | 冻结金额 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | Stock | 类解释 | 个股信息 | |
| 属性 | | 类型 | | 备注 |
| stockId | | int | | 个股id |
| stockName | | String | | 个股简称 |
| shareNapital | | double | | 股本 |
| earningsPerShare | | double | | 每股收益 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | StockHistory | 类解释 | 个股历史信息 | |
| 属性 | | 类型 | | 备注 |
| stockHistoryId | | int | | 个股历史信息id |
| stockId | | int | | 股票id |
| date | | date | | 日期 |
| openPrice | | double | | 开盘价 |
| closePrice | | double | | 收盘价 |
| highestPrice | | double | | 最高价 |
| lowestPrice | | double | | 最低价 |
| volume | | int | | 成交量 |
| increase | | double | | 涨幅 |
| amplitude | | double | | 振幅 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | HoldPosition | 类解释 | 持仓股票 | |
| 属性 | | 类型 | | 备注 |
| holdPositionId | | int | | id |
| userId | | int | | 用户id |
| stockId | | int | | 股票id |
| stockName | | String | | 个股简称 |
| costPrice | | double | | 成本价 |
| positionNumber | | int | | 持仓数量 |
| availableNumber | | int | | 可用数量 |
| openingTime | | date | | 建仓时间 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | SelfSelectStock | 类解释 | 自选股 | |
| 属性 | | 类型 | | 备注 |
| selfSelectStockId | | int | | id |
| userId | | int | | 用户id |
| stockId | | int | | 股票id |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | HistoryHoldPosition | 类解释 | 历史持仓 | |
| 属性 | | 类型 | | 备注 |
| historyHoldPositionId | | int | | id |
| userId | | int | | 用户id |
| stockId | | int | | 股票id |
| stockName | | String | | 股票简称 |
| buildPositionDate | | date | | 建仓时间 |
| clearPositionDate | | date | | 清仓时间 |
| stockHoldDay | | int | | 持股天数 |
| totalProfitAnd\_loss | | double | | 总盈亏 |
| totalPurchaseAmount | | double | | 总买入金额 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | Order | 类解释 | 委托买/卖单 | |
| 属性 | | 类型 | | 备注 |
| orderId | | int | | id |
| stockId | | int | | 股票id |
| type | | int | | 类型 |
| userId | | int | | 用户id |
| localTime | | time | | 本机时间 |
| date | | date | | 日期 |
| tradeStraregy | | int | | 策略 |
| orderAmount | | int | | 委托数量 |
| exchangeAmount | | int | | 成交数量 |
| orderPrice | | double | | 委托价格 |
| exchangeAveragePrice | | double | | 成交均价 |
| cancelNumber | | int | | 撤销数量 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | TradeOrder | 类解释 | 交易单 | |
| 属性 | | 类型 | | 备注 |
| tradeOrderId | | int | | id |
| stockId | | int | | 股票id |
| buyOrderId | | int | | 委托买单id |
| sellOrderId | | int | | 委托卖单id |
| sellPoint | | int | | 卖方完成标识位 |
| bugPoint | | int | | 买方完成标识位 |
| time | | time | | 时间 |
| date | | date | | 日期 |
| tradePrice | | double | | 成交价 |
| exchangeAmount | | int | | 成交数量 |
| tradeType | | int | | 买/卖盘 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | TransactionOrder | 类解释 | 历史成交单 | |
| 属性 | | 类型 | | 备注 |
| transactionOrderId | | int | | id |
| stockId | | int | | 股票id |
| sellId | | int | | 卖买标识 |
| sellOrderId | | int | | 委托单id |
| ownerId | | int | | 拥有者id |
| time | | time | | 时间 |
| date | | date | | 日期 |
| tradePrice | | double | | 成交价 |
| exchangeAmount | | int | | 成交数量 |
| tradeType | | int | | 买/卖盘 |
| serviceTax | | double | | 手续费 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | News | 类解释 | 公告 | |
| 属性 | | 类型 | | 备注 |
| newsId | | int | | id |
| newsType | | String | | 公告类型 |
| newsContent | | String | | 公告内容 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | Help | 类解释 | 指南 | |
| 属性 | | 类型 | | 备注 |
| helpId | | int | | id |
| name | | String | | 名称 |
| type | | String | | 类型 |
| content | | text | | 内容 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | Information | 类解释 | 消息提醒 | |
| 属性 | | 类型 | | 备注 |
| informationId | | int | | id |
| userId | | int | | 用户id |
| time | | time | | 时间 |
| date | | date | | 日期 |
| content | | String | | 内容 |

# 接口设计(Controller)

## 1.个人中心（UserController）

### 新用户注册

| 功能名称 | 新用户注册 | | |
| --- | --- | --- | --- |
| 接口方法 | register() | | |
| URL地址 | /api/register | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userName | String | 用户名 | Y |
| email | String | 邮箱地址 | Y |
| verification | Int | 验证码 | Y |
| loginPassword | String | 登录密码 | Y |
| transactionPassword | Int | 交易密码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200（成功）  10000(失败：未知的错误)  10001(失败：用户名重复)  10001（失败：邮箱已被注册）  10002（失败：验证码错误） | Y |
| message | String | 成功或失败的消息 | Y |

### 获取验证码

| 功能名称 | 获取验证码 | | |
| --- | --- | --- | --- |
| 接口方法 | getValidateCode() | | |
| URL地址 | /api/validateCode | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| email | String | 邮箱地址 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200（成功）  10000(失败：未知的错误)  50001（失败：获取验证码失败） | Y |
| message | String | 成功或失败的消息 | Y |
|  |  |  |  |

### 用户登录

| 功能名称 | 用户登录 | | |
| --- | --- | --- | --- |
| 接口方法 | Login | | |
| URL地址 | /api/login | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userName | String | 用户名 | Y |
| loginPassword | String | 登录密码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10002(失败：登录用户名错误)  10002(失败：登录密码错误) | Y |
| message | String | 成功或失败的消息 | Y |

### 找回密码

| 功能名称 | 找回密码 | | |
| --- | --- | --- | --- |
| 接口方法 | findPassword() | | |
| URL地址 | /api/findPassword | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| email | String | 邮箱地址 | Y |
| verification | Int | 验证码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  50004(失败：邮箱不存在)  10002(失败：验证码错误) | Y |
| message | String | 成功或失败的消息 | Y |
| data | 对象 | 包含邮箱、用户名、验证码 | Y |

| 功能描述 | Data |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | | 参数类型 | 参数说明 | 是否必须 |
| userID | | String | 用户名 | Y |

### 重置密码

| 功能名称 | 重置密码 | | |
| --- | --- | --- | --- |
| 接口方法 | resetPassword() | | |
| URL地址 | /api/resetPassword | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userName | String | 用户名 | Y |
| email | String | 邮箱地址 | Y |
| loginPaeeword | String | 登录密码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200（成功）500109(失败) | Y |
| message | String | 成功或失败的消息 | Y |

### 资产展示

| 功能名称 | 用户资金 | | |
| --- | --- | --- | --- |
| 接口方法 | getFunds() | | |
| URL地址 | /api /getFunds | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | int | 用户代码 | Y |
|  |  |  |  |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| code | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：要获取的对象不存在） | Y |
| message | String | 成功或失败的消息 | Y |
| data | 对象 | 用户资产信息 | Y |

| 功能描述 | 用户资产信息 |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | | 参数类型 | 参数说明 | 是否必须 |
| totalFunds | | Double | 总资产 | Y |
| holdPosProAndLos | | Double | 持仓盈亏 | Y |
| availableFunds | | Double | 可用资金 | Y |
| totalMarketValue | | Double | 总市值 | Y |
| todayProAndLos | | Double | 当日盈亏 | Y |
| freezFunds | | Double | 冻结资金 | Y |

### 消息提醒

| 功能名称 | 消息提醒 | | |
| --- | --- | --- | --- |
| 接口方法 | getInformation() | | |
| URL地址 | /api/getInformation | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | int | 用户代码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | int | 200(成功)  10000(失败：未知的错误)  10003(失败：消息不存在) | Y |
| message | String | 成功或失败的消息 | Y |
| data | String | 消息内容 | N |

| 对象名 | 消息内容 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| information | String | 消息内容 | Y |

## 2.信息展示（StockInformationController）

### 股票查询

| 功能名称 | 股票查询 | | |
| --- | --- | --- | --- |
| 接口方法 | getStockList() | | |
| URL地址 | /api/getStockList | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| searchStock | string | 查询股票  （若不设置，则显示所有股票） | N |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：对应股票不存在) | Y |
| message | String | “SUCCESS”表示股票信息获取成功  “FAIL” 表示股票信息获取失败 | Y |
| data | 数组 | 股票列表信息 | N |

| 数组名 | Data | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
| stockName | String | 股票名称 | Y |
| lastTradePrice | float | 最新成交价 | Y |
| increase | float | 今日涨幅 | Y |
| highestPrice | float | 最高价 | Y |
| lowestPrice | float | 最低价 | Y |
| todayOpenPrice | float | 今日开盘价 | Y |
| yesterdayOpenPrice | float | 昨日开盘价 | Y |
| turnover | float | 成交额 | Y |
| marketValue | float | 市值 | Y |
| cumulativeIncrease | float | 累计涨幅 | Y |
| priceEarningsRatio | float | 市盈率 | Y |
| priceToBookRatio | float | 市净率 | Y |

### K线图

| 功能名称 | K线图展示 | | |
| --- | --- | --- | --- |
| 接口方法 | KlineDiagramDisplay() | | |
| URL地址 | /api/KlineDiagramDisplay | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
| lineIdentifier | int | 0代表日线  1代表周线  2代表月线  默认为日线 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | int | 200(成功)  10000(失败：未知的错误)  10003(失败：对应股票不存在) | Y |
| message | string | 成功或失败的消息 | Y- |
| data | 对象 | 股票的基本数值 | - |

| 对象名 | Data | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| data | string | 日期  （周线图展示周五，月线图展示每个月最后一天） | Y |
| weekId | int | 周几  1代表周一 | Y |
| openPrice | float | 开盘价 | Y |
| closePrice | float | 收盘价 | Y |
| highestPrice | float | 最高价 | Y |
| lowestPrice | float | 最低价 | Y |
| increase | float | 涨幅 | Y |
| amplitude | float | 振幅 | Y |
| volume | int | 成交量 | Y |

### 分时图

| 功能名称 | 分时图展示 | | |
| --- | --- | --- | --- |
| 接口方法 | timeSharingDisplay() | | |
| URL地址 | /api/timeSharingDisplay | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
|  |  |  |  |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | int | 200(成功)  10000(失败：未知的错误)  10003(失败：对应股票不存在) | Y |
| message | string | 成功或失败的消息 | Y |
| data | 对象 | 股票的基本数值 | Y |

| 对象名 | 股票信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
| stockName | string | 股票简称 | Y |
| realTime | string | 时间 | Y |
| latestPrice | float | 最新价 | Y |
| averagePrice | float | 均价 | Y |
| volume | float | 成交量 | Y |

### 实时信息

| 功能名称 | 实时更新的数值 | | |
| --- | --- | --- | --- |
| 接口方法 | realTimeDataDisplay() | | |
| URL地址 | /api/realTimeDataDisplay | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | int | 200(成功)  10000(失败：未知的错误)  10003(失败：对应股票不存在) | Y |
| message | string | 成功或失败的消息 | Y |
| data | 对象 | 股票的基本数值 | Y |
|  |  |  |  |

| 对象名 | 股票信息 | | | |
| --- | --- | --- | --- | --- |
| 参数名 | 参数类型 | | 参数说明 | 是否必须 |
| stockID | int | | 股票代码 | Y |
| stockName | string | | 股票简称 | Y |
| weiBi  commissionProportion | float | | 委比 | Y |
| commissionPrice | list | price | 委买 | Y |
| count | 委卖 | Y |
| openPrice | float | | 开盘价 | Y |
| highestPrice | float | | 最高价 | Y |
| lowestPrice | float | | 最低价 | Y |
| latestPrice | float | | 最新价 | Y |
| upsAndDowns | float | | 涨跌 | Y |
| increase | float | | 涨幅 | Y |
| outMarket | float | | 外盘 | Y |
| inMarket | float | | 内盘 | Y |
| conversionHand | float | | 换手 | Y |
| totalMarketCapitalization | float | | 总市值 | Y |
| staticP/ERatio | float | | 静态市盈率 | Y |
| dailyLimit | float | | 涨停价 | Y |
| downLimitBoard | float | | 跌停价 | Y |
| cityNet; | float | | 市净值 | Y |

### 自选股

| 功能名称 | 自选股查看 | | |
| --- | --- | --- | --- |
| 接口方法 | selfSelectedStock() | | |
| URL地址 | /api/selfSelectedStock | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | int | 用户代码 | Y |
| stockID | int | 股票代码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | int | 200(成功)  10000(失败：未知的错误)  10003(失败：未查询到对应的股票列表) | Y |
| message | string | “SUCCESS”表示股票信息获取成功  “FAIL” 表示股票信息获取失败 | Y |
| data | 数组 | 股票列表信息 | N |

| 对象名 | 股票信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
| stockName | String | 股票名称 | Y |
| latestPrice | float | 最新成交价 | Y |
| increase | float | 涨幅 | Y |
| highestPrice | float | 最高价 | Y |
| lowestPrice | float | 最低价 | Y |
| todayOpenPrice | float | 今日开盘价 | Y |
| yesterdayOpenPrice | float | 昨日开盘价 | Y |
| turnover | float | 成交额 | Y |
| marketValue | float | 市值 | Y |
| cumulativeIncrease | float | 累计涨幅 | Y |
| priceEarningsRatio | float | 市盈率 | Y |
| priceToBookRatio | float | 市净率 | Y |

### 公告

| 功能名称 | 公告 | | |
| --- | --- | --- | --- |
| 接口方法 | getNews() | | |
| URL地址 | /api/getNews | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
|  |  |  |  |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：公告不存在) | Y |
| message | String | “SUCCESS”表示股票信息获取成功  “FAIL” 表示股票信息获取失败 | Y |
| data | String | 公告内容 | N |

| 对象名 | 公告内容 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| News | Int | 公告的内容 | Y |

## 3.信息统计(TradeInformationController)

### 当前持仓

| 功能名称 | 当前持仓 | | |
| --- | --- | --- | --- |
| 接口方法 | presentHoldPositionInfo() | | |
| URL地址 | /api/presentHoldPositionInfo | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | Int | 用户代码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：当前持仓信息不存在) | Y |
| message | String | 返回是否成功信息 | Y |
| data | 数组 | 当前持仓列表 | N |

| 对象名 | 当前持仓信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | Int | 股票代码 | Y |
| stockName | String | 股票简称 | Y |
| increase | Double | 本日涨幅 | Y |
| presentPrice | Double | 现价 | Y |
| costPrice | Double | 成本价 | Y |
| actualAmont | Int | 实际数量 | Y |
| todayProfitAndLoss | Double | 当日盈亏 | Y |
| totalProfitAndLoss | Double | 总盈亏 | Y |
| profitAndLossRatio | Double | 盈亏比例 | Y |
| marketValue | Double | 市值 | Y |
| positionRatio | Double | 仓位占比 | Y |

### 持仓历史

| 功能名称 | 持仓历史 | | |
| --- | --- | --- | --- |
| 接口方法 | historyHoldPositionInfo() | | |
| URL地址 | /api/historyHoldPositionInfo | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | Int | 用户代码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：持仓历史信息不存在) | Y |
| message | String | 成功或失败的消息内容 | Y |
| data | 数组 | 历史持仓信息列表 | N |

| 对象名 | 历史持仓信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| stockID | int | 股票代码 | Y |
| stockName | string | 股票简称 | Y |
| buildPositionDate | string | 建仓日期 | Y |
| clearPositionDate | string | 清仓日期 | Y |
| holdStockDays | int | 持股天数 | Y |
| totalProfitAndLoss | double | 总盈亏 | Y |
| profitAndLossRatio | double | 盈亏比例 | Y |

### 历史成交

| 功能名称 | 历史成交 | | |
| --- | --- | --- | --- |
| 接口方法 | historyExchangeInfo() | | |
| URL地址 | /api/historyExchangeInfo | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | Int | 用户名 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：历史成交信息不存在) | Y |
| message | String | 成功或失败的消息内容 | Y |
| data | 数组 | 历史成交列表 | N |

| 对象名 | 历史成交信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| exchangeDate | string | 成交日期 | Y |
| exchangeTime | string | 成交时间 | Y |
| stockID | Int | 股票代码 | Y |
| stockName | String | 股票简称 | Y |
| operation | String | 操作（买入/卖出） | Y |
| exchangeAmount | Int | 成交数量 | Y |
| exchangeAveragePrice | Double | 成交均价 | Y |
| exchangeAmount | Double | 成交金额 | Y |
| stockBalance | Int | 股票余额 | Y |
| contractNumber | Int | 合同编号 | Y |
| exchangeNumber | Int | 成交编号 | Y |
| serviceTax | Double | 手续费 | Y |
| stampTax | Double | 印花税 | Y |
| incidentalCharge | Double | 其他杂费 | Y |
| actualAmount | Double | 发生金额（实际金额） | Y |
| market | String | 交易市场 | Y |
| revokeAmount | Int | 撤单数量 | Y |

### 当日成交

| 功能名称 | 当日成交 | | |
| --- | --- | --- | --- |
| 接口方法 | todayExchange() | | |
| URL地址 | /api/todayExchange | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | Int | 用户名 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：当日成交信息不存在) | Y |
| message | String | 成功或失败的消息内容 | Y |
| data | 数组 | 当日成交列表 | N |

| 对象名 | 当日成交信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| exchangeTime | string | 成交时间 | Y |
| stockID | Int | 股票代码 | Y |
| stockName | String | 股票简称 | Y |
| operation | String | 操作（买入/卖出） | Y |
| exchangeAmount | Int | 成交数量 | Y |
| exchangeAveragePrice | Double | 成交均价 | Y |
| exchangeAmount | Double | 成交金额 | Y |
| stockBalance | Int | 股票余额 | Y |
| contractNumber | Int | 合同编号 | Y |
| exchangeNumber | Int | 成交编号 | Y |
| serviceTax | Double | 手续费 | Y |
| stampTax | Double | 印花税 | Y |
| incidentalCharge | Double | 其他杂费 | Y |
| actualAmount | Double | 发生金额（实际金额） | Y |
| revokeAmount | Int | 撤单数量 | Y |

### 当日委托

| 功能名称 | 当日委托 | | |
| --- | --- | --- | --- |
| 接口方法 | todayOrder() | | |
| URL地址 | /api/todayOrder | | |
| 请求方式 | GET | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | Int | 用户代码 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：当日委托信息不存在) | Y |
| message | String | 成功或失败的消息内容 | Y |
| data | 数组 | 当日委托列表 | N |

| 对象名 | 当日委托信息 | | |
| --- | --- | --- | --- |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| orderDate | string | 委托日期 | Y |
| orderTime | string | 委托时间 | Y |
| stockID | Int | 股票代码 | Y |
| stockName | String | 股票简称 | Y |
| operation | String | 操作（买入/卖出） | Y |
| orderAmount | Int | 委托数量 | Y |
| exchangeAmount | Int | 成交数量 | Y |
| exchangeAveragePrice | Double | 成交均价 | Y |
| orderPrice | Double | 委托价格 |  |
| revokeAmount | Int | 撤单数量 | Y |
| orderType | String | 订单类型（市价委托/限价委托） | Y |
| contractNumber | Int | 合同编号 | Y |

## 买卖交易(TradeController)

### 买入/卖出

| 功能名称 | 限价买入 | | |
| --- | --- | --- | --- |
| 接口方法 | buyOrSale() | | |
| URL地址 | /api/buyOrSale | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| userID | Int | 委托人id | Y |
| stockID | String | 股票id | Y |
| isSaleOrBuy  is | Int | 买卖标识 | Y |
| orderCount | Int | 委托数量 | Y |
| orderPrice | Float | 委托价格 | N |
| orderType | String | 委托类型 | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：对应股票不存在)  50005(失败：资金不足)  50007(失败：非法的交易时间) | Y |
| message | String | 成功或失败的消息内容 | Y |
| data |  | 无 | N |

### 撤单

| 功能名称 | 撤单 | | |
| --- | --- | --- | --- |
| 接口方法 | cancelTheOrder() | | |
| URL地址 | /api/cancelTheOrder | | |
| 请求方式 | POST | | |
| 请求参数 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| data | Int | 委托单id | Y |
| 接口返回 | | | |
| 参数名 | 参数类型 | 参数说明 | 是否必须 |
| resultCode | Int | 200(成功)  10000(失败：未知的错误)  10003(失败：对应股票不存在)  50006(失败：委托已完成)  50007(失败：非法的交易时间) | Y |
| message | String |  | Y |
| data |  | 无 | N |

# Service设计

## 模块名：信息管理

### 功能名：历史成交单

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | TransactionService | 类解释 | 历史成交业务逻辑类 |
| 方法名 | | 方法说明 | |
| public List<TransactionOrder> getTransactionOrder (Date start,Date end,List<TransactionOder >) | | 根据用户选择的起始日期和结束日期选出符合条件的股票，返回历史成交单列表，每个成交单包含的信息：成交日期、成交时间、证券代码、证券名称、操作、成交数量、成交均价、成交金额、股票金额、合同编号、成交编号、手续费、印花税、其他杂费、发生金额、交易市场、股东账户，撤单数量 | |
| public List<TransactionOrder> getTranscationOrder(List<TransactionOrder >) | | 用户全部历史成交单，参数：成交单列表。返回历史成交单列表，每个成交单包括成交日期、成交时间、证券代码、证券名称、操作、成交数量、成交均价、成交金额、股票金额、合同编号、成交编号、手续费、印花税、其他杂费、发生金额、交易市场、股东账户，撤单数量 | |

### 功能名：历史持仓

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | HistoryHoldPositionService | 类解释 | 历史持仓业务逻辑类 |
| 方法名 | | 方法说明 | |
| public List<HistoryHoldPosition> getHistoryHoldPosition(Date start,Date end,List<HistoryHoldPosition>) | | 根据用户选择的起始日期和结束日期选出符合条件的股票，返回股票列表，每只股票包含的信息：明细、证券代码、证券名称、建仓日期、清仓日期、持股天数、总盈亏、盈亏比例。 | |
| public List<HistoryHoldPosition> getHistoryHoldPosition(List<HistoryHoldPosition>) | | 用户全部历史持仓信息，每只股票包含的信息：明细、证券代码、证券名称、建仓日期、清仓日期、持股天数、总盈亏、盈亏比例。 | |

## 模块名：行情展示

### 功能名：K线图展示

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | StockHistoryService | 类解释 | K线图的展示 |
| 方法名 | | 方法说明 | |
| public List<StockRepository> getAllStockRepority(int stock\_id) | | 接收股票代码  返回数据库的该代码的历史信息list | |
| public List<KlineDataVO> as(int stock\_id) | | 接收股票代码，向controller返回前端股票所需要的所有信息 | |
| public int CalculateDayOfTheWeek(Date date) | | 输入日期，返回这天是星期几 | |

## 模块名：股票交易

### 功能名：买卖委托

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | OrderService | 类解释 | 处理下达限价买入的service |
| 方法名 | | 方法说明 | |
| public int buyLimitOrder(OrderVO order) | | 该方法处理用户下达限价买入的申请，首先service调用repository计算用户可用余额，判断该余额是否大于限价买入委托单的总额，如果小于，返回余额不足错误信息；否则，调用repository将该订单信息存入数据库，将订单传入撮合系统。 | |
| public int buyMarketOrder(OrderVO order) | | 该方法处理用户下达市价买入的申请，首先service判断该系统状态，如果是闭市，设置冻结金额为前收盘价的125%，否则为涨停价；调用repository计算用户可用余额，判断该余额是否大于冻结金额，如果小于，返回余额不足错误信息；否则，调用repository将该订单信息存入数据库，将订单传入撮合系统。 | |
| public int sellLimitOrder(OrderVO order) | | 该方法处理用户下达市价买入的申请，首先service调用repository计算用户可卖出股票，判断该数量是否大于订单数量，如果小于，返回股票不足错误信息；否则，调用repository将该订单信息存入数据库，将订单传入撮合系统。 | |
| public int sellMarketOrder(OrderVO order) | | 该方法处理用户下达市价买入的申请，首先service调用repository计算用户可卖出股票，判断该数量是否大于订单数量，如果小于，返回股票不足错误信息；否则，调用repository将该订单信息存入数据库，将订单传入撮合系统。 | |

### 功能名：撤单

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | CancelTheOrderService | 类解释 | 处理撤单申请 |
| 方法名 | | 方法说明 | |
| public boolean applyCancelOrder(int orderId) | | 该方法向撮合系统发出撤单申请。 | |

# repository设计

## 模块名：信息管理

### 功能名：历史成交单

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | TransctionRepository | 类名解释 | 历史成交数据访问类 |
| 方法名 | | 方法说明 | |
| public List<Tra  nsctionOrder > getTransactionOrder (String useId) | | 在数据库表transaction\_order中找到买方id与用户id相同的成交单，并为每个成交单添加一个属性“买入”。  在数据库表transaction\_order中找到卖方id与用户id相同的成交单，并为每个成交单添加一个属性“卖出”。  将这些成交单整合为一个列表。 | |

### 功能名：历史持仓

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | HistoryHoldPositionPepository | 类名解释 | 历史持仓数据访问类 |
| 方法名 | | 方法说明 | |
| public List<HistoryHoldPosition> getHistoryHoldPosition (String userName) | | 根据用户名从数据库中获取该用户的历史持仓的股票信息（将数据库中的表history\_hold\_position和表stock按照stock\_id合并,返回history\_hold\_position表中出现的股票信息，每只股票包含的信息：明细、证券代码、证券名称、建仓日期、清仓日期、持股天数、总盈亏、盈亏比例） | |

## 模块名：行情展示

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | public interface StockHistoryRepository extends JpaRepository<StockRepository, Integer> | 类名解释 | K线图数据库查询 |
| 方法名 | | 方法说明 | |
| public List<StockRepository> findByStock\_id(int Stock\_id); | | 查询数据库，输入股票ID，返回股票信息列表 | |

## 模块名：股票交易

### 功能名：下达委托

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 | OrderRepository | 类名解释 | 处理用户订单的repository |
| 方法名 | | 方法说明 | |
| public double getAvailableBalance(int id) | | 查询该用户user\_fund资金账号表中资金余额和当日委托冻结金额，返回可用余额。 | |
| public Boolean insertOrder(Order order) | | 将委托单存入数据库。 | |

# 关键模块时序图

## 7.1 登录

## 7.2 买卖

# 竞价算法设计

## 8.1 集合竞价算法

按照价格优先时间优先的原则，建立买单、卖单队列

买单的优先度是价格，卖单的优先度是价格的负数

队列使用TreeMap<Priority,PriceLeader>进行维护,key是优先度，PricaLeader是价格挡位

PriceLeader中包含Order的时间优先队列

插入的先后自动根据时间排列

首先计算成交价

1、如果买单或者卖单队列为空，返回；

2、获取买单、卖单队列对头数据

如果买单对头价格小于卖单对头价格，返回；

3、使用买单吃卖单的策略，

确定成交量，成交量 = min（买单剩余数量，卖单剩余数量）

总成交量 = 总成交量 + 成交量

更新买单剩余数量 = 买单剩余数量 - 成交量

更新卖单剩余 = 卖单剩余数量 - 成交量

更新最新买单价格 = 买单委托价

更新最新卖单价格 = 卖单委托价

4、如果买单剩余为0，获取下一买单

如果该买单为空，跳转；

5、如果卖单剩余为0，获取下一卖单

如果该卖单为空，跳转

6、如果买单价格大于卖单价格，重复3456

7、如果最新买单价格等于最新卖单价格，最终成交价为该价格，算法结束；

如果不相等：

深市：计算P1 = |大于买单最新价格的买单数量 - 小于买单最新价格的卖单数量|

P2 = |大于卖单最新价格的买单数量 - 小于卖单最新价格的卖单数量|

如果P1<P2 成交价 = 买单最新价格

如果P1>P2 成交价 = 卖单最新价格

否则 开盘成交价 = 上一日收盘价

收盘成交价 = 最新成交价

沪市：取两个价格的均价，算法结束。

算法返回 成交价 成交数量等。若没有发生撮合，返回NULL。

## 8.2 正式撮合

1、如果成交量小于等于0，算法结束。

2、设定剩余成交量 = 总成交量

3、如果剩余成交量 > 0

取出下一个非撤单的买单和卖单，成交量为委托量较小的一个，成交价为撮合成交价。

剩余成交量 = 剩余成交量 - 成交量

修改两委托剩余成交量 = 原剩余成交量 - 成交量

记录这笔成交

将成交量不为0的订单再次入队

循环操作

4、将成交记录返回。

## 8.3 连续竞价

订单来临时，修改该订单剩余委托量为订单委托量

成交价 = 对手方委托价格

判断订单类型

如果是限价交易，直接撮合，撮合失败或撮合剩余加入队列

否则继续判断：

深市：

最优五档即时成交剩余撤单：

只对对手方五档价格进行撮合，还有剩余撤销；未成交直接撤单

最优五档即时成交剩余转限价：

只对对手方五档价格进行撮合，剩余转限价，限价金额为第一笔成交价；未成交直接撤单

对手方最优价格

设置限价价格为对手方一档价格，进行撮合，剩余转限价；未成交直接撤单

本方最优价格

设置限价价格为本方一档价格，进行撮合，剩余转限价；未成交直接撤单

即时成交剩余撤单：

按对手方申报队列撮合，还有剩余撤单；未成交直接撤单

沪市：

最优五档即时成交剩余撤单：

只对对手方五档价格进行撮合，还有剩余撤销；未成交直接撤单

全额成交或撤销委托：

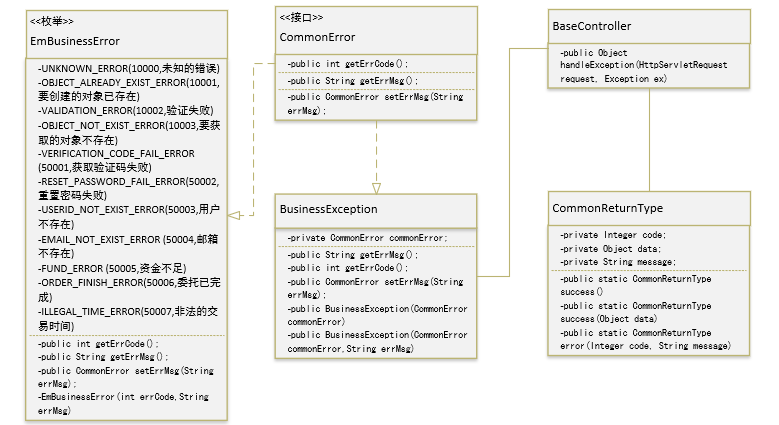
按对手方申报队列撮合，如果还有剩余则不成交直接撤销委托

将成交信息返回。

# 信息返回格式设计-Json格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | CommonReturnType | 类解释 | 通用返回格式类 | |
| 属性 | | 类型 | | 备注 |
| Code | | int | | 状态码 |
| Message | | string | | 内容 |
| Data | | Object | | 数据 |

# 异常错误模块设计



错误代码：

-UNKNOWN\_ERROR(10000,未知的错误)  
-OBJECT\_ALREADY\_EXIST\_ERROR(10001,要创建的对象已存在)  
-VALIDATION\_ERROR(10002,验证失败)  
-OBJECT\_NOT\_EXIST\_ERROR(10003,要获取的对象不存在)  
-VERIFICATION\_CODE\_FAIL\_ERROR(50001,获取验证码失败)  
-RESET\_PASSWORD\_FAIL\_ERROR(50002, 重置密码失败)  
-USERID\_NOT\_EXIST\_ERROR(50003,用户不存在)  
-EMAIL\_NOT\_EXIST\_ERROR(50004, 邮箱不存在)  
-FUND\_ERROR(50005,资金不足)  
-ORDER\_FINISH\_ERROR(50006,委托已完成)  
-ILLEGAL\_TIME\_ERROR(50007,非法的交易时间)