



DataVest 量化交易系统

项目报告

目录

术语	4
1 需求概述	6
1.1 需求背景	6
1.2 需求列表	6
1.3 需求范围及限制	7
1.3.1 数据来源限制	7
1.3.2 技术限制	7
1.3.3 法律法规限制	7
1.3.4 资金限制	7
2 需求分析	8
2.1 竞品分析	8
2.1.1 部分主流平台对比	8
2.1.2 实现原理	10
2.1.3 与本产品优劣对比	11
2.1.4 结论与启示	11
2.2 功能性需求	12
2.3 设计约束	13
2.3.1 技术约束	13
2.3.2 业务约束	14
2.3.3 用户约束	14
2.4 依赖分析	15
2.5 影响分析	16
2.5.1 公共框架、控件相关软件影响分析	16
2.5.2 对用户的影响分析	16
3 系统方案设计	17
3.1 零层设计	17
3.2 一层设计	18
3.2.1 方案选型	18
3.2.2 整体框图	19
3.2.3 模块设计	20

3.2.4 关键流程设计	20
3.3 二层设计	20
3.4 技术方案风险点	26
4 数据结构设计	27
4.1 主要数据结构	27
4.1.1 数据库设计	28
5 接口设计	29
5.1 提供接口	29
5.2 依赖接口	32
6 系统测试	33
6.1 概述	33
6.2 测试范围	33
6.3 单元测试 V1	33
6.3.1 V1.1 白盒测试	33
6.3.2 V1.2 黑盒测试	34
6.3.3 V1.3 白盒测试	35
6.3.4 V1.4 黑盒测试	36
6.3.5 V1.5 白盒测试	37
6.3.6 V1.6 黑盒测试	38
6.3.7 V1.7 黑盒测试	39
6.4 系统测试 V2	40
6.4.1 V2.1 黑盒测试	40
6.4.2 V2.2 黑盒测试	41

术语

1. 回测：根据历史数据来验证交易策略的可行性和有效性的过程。
2. 复权：是指在股票交易中，对历史股价数据进行调整，以反映股票分红、配股等因素对股价的影响。通常会将历史股价进行复权，使得历史数据更加准确和可比较。
3. 回测滑点：交易中理想成交价和实际成交价的差异。
4. 夏普比率 = (投资组合年化收益率 - 年化无风险利率) / 投资组合的标准差。夏普比率的分子是超额收益，分母是波动风险，它能够告诉我们“每承担一份风险，可以获得多少超额回报”。
5. 索提诺比率：索提诺比率是一种衡量投资组合相对表现的方法。与夏普比率有相似之处，但索提诺比率运用下偏标准差而不是总标准差，以区别不利和有利的波动。这一比率越高，表明基金承担相同单位下行风险能获得更高的超额回报率。
6. 最大回撤率：在选定周期内任一历史时点往后推，产品净值走到最低点时的收益率回撤幅度的最大值。最大回撤用来描述买入产品后可能出现的最糟糕的情况。
7. 莫伦卡选股策略：莫伦卡根据四个指标，选出有价值的股票所用的策略。他分别关注：1) 市盈率（市值/净利润），一般情况下，一只股票市盈率越低，市价相对于股票的盈利能力越低，表明投资回收期越短，投资风险就越小，股票的投资价值就越大；反之则结论相反。这里选择市盈率低于 30，且大于 0 的。2) 基于历年年报数据，过去 5 年平均净资产收益率高于 14%。3) 经营现金流为正。4) 新期的净利润大于前 5 年的净利润
8. MACD 金叉死叉趋势跟踪策略：通过观察 MACD 线和信号线的交叉来判断市场趋势的转变。当 MACD 线由下向上穿过信号线时，为金叉信号，表明市场可能转为上涨趋势，此时买入；而当 MACD 线由上向下穿过信号线时，为死叉信号，表明市场可能转为下跌趋势，此时卖出。
9. MACD 与 HMA 结合策略：MACD 和 HMA（赫尔移动平均线）的混合旨在通过减少滞后和提高响应能力来完善基本 MACD 移动平均线策略。这种组合充分利用了这两个指标的优势。这里使用的 HMA 均线是 10 周期和 20 周期 HMA。就像 MACD 一样，交易者寻找交叉事件，但在这种情况下，交叉事件来自 MACD 和 HMA，并且最好彼此非常接近。

对于看涨前景，交易者可能会在 10 周期 HMA 穿越 20 周期 HMA 的同时，寻找 MACD 线穿越信号线上方。交易者经常将止损设置在附近波动点之上或之下。

表 1: 缩略语清单

缩略语	全称	说明
MACD	异同移动平均线 Moving Average Convergence / Divergence	利用收盘价的短期（常用为 12 日） 指数移动平均线与长期（常用为 26 日）指数移动平均线之间的聚合与分 离状况，对买进、卖出时机作出研判 的技术指标。
HMA	赫尔移动平均线 Hull Moving Average	通常被视为早期捕捉市场趋势的有效 工具，为交易者提供交易时机的优 势。它结合了简单移动平均线 (SMA) 和加权移动平均线 (WMA) 的平滑优势，以实现价格变化的更 快响应。
UPI	Ulcer 绩效指数 Ulcer Performance Index	是一种衡量投资组合风险的指标，常 用于策略回测和投资评估中。它通过 测量资产价格下跌的幅度和持续时间 来评估投资组合的波动性和风险水 平。

1 需求概述

1.1 需求背景

随着全球金融市场的不断发展和扩大，市场中的交易品种、交易量和交易参与者都在不断增加。市场结构和价格行为变得更加复杂，金融市场中的信息量庞大且实时变化。而且，如今金融市场不确定性逐渐加大，投资风险加剧。为了在这种环境下获得投资竞争优势，越来越多的金融机构和投资者开始采用量化交易策略，通过算法和自动化交易来提高交易效率和盈利能力。

面对这种需求，我们的产品要从海量数据中提取有价值的信息，具有高效的数据处理、分析和交易执行，帮助投资者发现投资机会并迅速作出交易决策。

同时，为了应对市场不断加剧的不确定性和风险，我们通过风险模型和算法来制定相应的风险控制策略。在交易过程中可以自动执行止损、止盈等风险管理操作，降低人为干预的风险，并尽可能确保投资组合的风险水平符合预设的要求。

我们的产品核心功能是实现完全自动化的交易执行，通过预设的交易策略和规则，自动完成买卖订单的生成、执行和监控。这种自动化交易可以大大提高交易效率，降低交易成本，并减少人为错误的发生。

1.2 需求列表

表 2: 功能性需求列表

功能性需求	需求说明
数据获取和处理	能够方便地接入各种金融数据源，例如 Alpha Vantage API 等，并具有强大的数据处理能力，支持数据的清洗、转换和分析
策略研究和开发	基于 python 开发，提供丰富的数学和统计工具，支持策略的回测和优化，以便用户能够灵活地实现自己的交易策略
风险管理和监控	内置风险管理和监控工具，能够实时评估和管理交易头寸的风险，包括市场风险、信用风险和操作风险等
可视化工具	提供直观的数据可视化工具，帮助用户更好地理解数据和策略的表现
技术支持和社区	提供专业的技术支持，同时拥有活跃的用户社区，方便用户之间的交流和协作

表 3: 非功能性需求列表

非功能性需求	需求说明
高效性与可用性	具有低延迟的交易执行能力，支持算法交易，保证系统随时可用
合规性与安全性	确保平台符合相关金融监管要求，如《证券法》、《期货交易管理条例》、《反洗钱法》、《数据安全法》等，保证交易数据的安全性和隐私性
扩展性	支持系统功能和容量的灵活扩展，适应不断增长的用户和交易量。

1.3 需求范围及限制

1.3.1 数据来源限制

在设计金融交易系统时，首先需要考虑可用的数据源和数据获取方式。这包括但不限于市场行情数据、交易所数据、财务报表数据等。数据的来源可能涉及不同的供应商或数据提供者，因此需要评估其可靠性、实时性和数据质量等因素。同时，还需要考虑到数据获取可能存在的延迟和不确定性，这对于决策和交易执行的时效性至关重要。

1.3.2 技术限制

在考虑 *Python* 作为编程语言时，需要考虑其对系统性能和可维护性的影响。对于金融交易系统而言，性能和稳定性是至关重要的，因此需要选择高效可靠的技术方案。同时，还需要考虑技术的适应性和扩展性，以应对未来可能的需求变化和系统扩展。

1.3.3 法律法规限制

金融市场受到严格的监管，交易行为必须遵守相关的法律法规。在设计交易系统时，需要确保系统的合法性和规范性，以免触犯监管机构的规定。因此，在系统设计阶段就需要充分考虑法律法规对系统架构、交易流程和数据安全等方面的要求。

1.3.4 资金限制

考虑交易系统的资金规模和交易成本是设计过程中必不可少的一部分。系统设计应该考虑交易规模的扩展性和风险控制的能力，以确保交易系统能够适应不同规模的资金投入。同时，需要评估交易成本对于交易策略的影响，避免超出可承受的风险范围。资金限制也涉及到资金管理、风险控制和资产配置等方面的问题，在系统设计中需要全面考虑。

2 需求分析

2.1 竞品分析

2.1.1 部分主流平台对比

表 4: 主流平台对比

平台	迅投 QMT	恒生 PTrade	掘金量化	米筐	聚宽
数据	提供全部日/分钟/Tick 级别股票数据, 及财务、分红送配、行业、板块等数据。还提供期货、期权的连续数据及过期合约。	提供沪深股票、期权、期货等全品种 L2 行情支持。	提供近 10 年日/分钟/Tick 级别股票数据, 及财务、分红送配、行业、板块等数据。还提供股指期货、商品期货的连续数据。	提供股票、ETF、期货(股指、国债、商品期货)、现货的基本信息。股票、ETF 过去 10 多年以来每日市场数据, 股票、ETF 在 2005 年以来的分钟线数据。ETF 过去 20 多年以来的市场数据和财务数据。期货从 1999 年以来的每日行情数据。期货 2010 年以来的分钟线数据。中国 50 ETF、商品期权的日、分钟数据。舆情大数据。	提供 2005 年至今完整的股市 Level1 数据、上市公司财务数据、完整的停复权信息。实时更新行情数据, 盘后更新财务数据。此外还提供基金(包括 ETF、LOF、分级 A/B 基金、货币基金)的行情和净值数据, 金融期货数据、股票指数数据、行业板块数据、概念板块数据、宏观数据、行情数据等。
语言支持	支持 Python, VBA, 提供 API 接口。	支持 Python, 提供 API 接口。	支持 Python, Matlab, C, C++, C# 语言, 提供 API 接口。	提供基于 IPython Notebook 的研究平台, 支持 Python, Matlab, Excel。提供 API。	提供基于 IPython Notebook 的研究平台, 支持 Tick 级数据, 支持 Python2、Python3。提供 API。

续下页

续页

平台	迅投 QMT	恒生 PTrade	掘金量化	米筐	聚宽
回测	支持股票、期货、期权等品种回测及其混合回测，支持日、分钟、Tick 级全周期回测。	支持股票、可转债等品种回测，支持日、分钟、Tick 级全周期回测。	支持股票、期货等品种回测及其混合回测，支持日、分钟、Tick 级回测。	支持股票、ETF、期货等品种的回测，支持日、分钟级回测。	支持股票、基金、期货等品种的回测，支持日、分钟、Tick 级回测。
模拟交易	支持股票、期货、期权、可转债等品种的日、分钟级别的模拟交易。	支持股票、可转债、期权等品种的日、分钟级别的模拟交易。	支持股票、商品期货、股指期货等品种的日、分钟级别的模拟交易。	支持日、分钟级别的股票、ETF、期货等品种的模拟交易。	支持股票、股指期货、商品期货、ETF 等品种的日、分钟、Tick 级别的模拟交易。
实盘交易	支持股票、期货、两融、期权、可转债等量化实盘，实盘直接和对接券商开通，具有实盘交易权限后也可以手动交易。	支持股票、可转债、期权等量化实盘，有丰富的手工 T0 交易工具，实盘直接和对接券商开通，具有实盘交易权限后也可以手动交易。	支持股票、期货、两融等量化实盘，实盘开通需要客户提交申请和平台人工审核，具有实盘交易权限后也可以手动交易。	提供期货的实盘交易	和第一创业合作，支持股票、场内基金、期货的自动化实盘交易。
社区	券商专业客户服务群	券商专业客户服务群	掘金量化社区，活跃度一般	米筐量化社区”，活跃度较高。	“聚宽社区”，活跃度很高。
访问地址	合作券商提供	合作券商提供	www.myquant.com	www.ricequant.com	www.joinquant.com

2.1.2 实现原理

交易原理

- 全内存模型

并不需要多线程并发处理，而是依赖单线程无锁模型，让所有计算全部在内存中完成，处理速度可以获得 10 万的量级。订单的撮合和清算都在内存上进行完成。

使用全内存模型的好处除了极高的处理速度外，另一个好处是硬件成本很低，原因是内存非常便宜，单机 32G 内存就可以轻松实现每秒 10 万的订单处理速度。这里的每秒 10 万订单是指用户下单到订单撮合、清算全部完毕的整个流程，而不是单独指某个模块的处理速度。

不过，使用全内存模型的缺点是由于内存的易失性。因为不再通过数据库事务保证数据一致性，如果遭遇宕机、断电等系统故障，交易系统必须能可靠地恢复整个系统的状态，这对系统的可靠性提出了非常高的设计要求。

- 基于数据库的模型

如果把订单全部存入数据库，并且，每次撮合都基于数据库的订单进行排序，这种完全基于数据库操作，并通过数据库事务保证数据一致性的交易系统。

但是这样显然性能有限，并且，对数据库的硬件配置有非常高的要求。这种交易系统的处理能力每秒在 100 左右，提升系统性能完全依靠数据库服务器的硬件升级，成本高昂。

- 内存撮合 + 数据库清算

把订单从数据库转移到内存中，那么，在内存中维护两个有序表就非常快，因为每次撮合，系统只需要取有序表的前面若干条符合条件的订单，撮合速度有数量级的提升。

使用内存撮合的订单处理速度每秒可高达 100 万。然而，清算仍然是基于数据库事务，因此，整个系统的订单处理能力大约能提升到每秒 1000 单，比全部基于数据库处理的提高 10 倍左右。最终决定系统整体处理能力上限的是清算系统的处理速度。

策略的设计与研究通过理论分析研究市场理论，如有效市场假说、均值回归等，结合历史数据分析，通过分析历史市场数据，寻找潜在的市场规律和价格模式。然后识别影响资产价格的因素，如基本面因子、技术因子、市场情绪等。得到策略的研究结果。根据研究结果，设定策略的基本假设和条件。再设计具体的买卖规则，包括入场条件、出场条件、止损止盈等。最终根据市场类型和交易目标，设计趋势跟踪、反转交易、套利等不同类型的策略。

算法优化通过机器学习等方法优化交易算法, 提高交易效率和成功率。

页面设计原理

- **一致性:** 确保系统的交互元素和布局保持一致, 使用户能够快速熟悉
- **简洁性:** 界面简洁明了, 避免不必要的复杂性, 使用户能够直观地理解如何操作
- **反馈:** 为用户的每一个操作提供即时反馈, 让用户知道系统已经响应了他们的操作
- **易用性:** 设计时考虑到用户的学习曲线, 确保系统容易上手和使用

2.1.3 与本产品优劣对比

1. 本产品处于开发初期, 算法上可能稍逊于主流公司, 目前预计包括最基本的算法, 更加优化的有待研发。
2. 本产品页面设计尽可能追求用户友好, 但与较成熟的产品相比可能会有一些不足, 比如部分功能可能按键位置安排考量不足。
3. 功能种类上, 本产品可能有些没有考虑周全, 有所欠缺。
4. 社区功能可能相对不够完善。
5. 本产品目标尽可能开发一些相对独特的、与市面上有所不同的功能。

2.1.4 结论与启示

通过对市场上的竞品分析, 我们的系统要注重以下几个方面:

• 功能特色

比较各平台的功能模块、提供的工具和服务, 比如交易算法、数据分析、风险管理等, 我们发现了它们各自的优势和特点。我们要结合上面几家公司的产品的优点, 比如高效交易等, 来开发平台。

• 技术能力与框架

通过分析平台的技术架构、算法效率、系统稳定性、处理线程等方面, 可以评估各平台的技术实力和服务能力。

• 用户体验

用户界面设计、操作便捷性、客户服务质量等都是衡量用户体验的重要因素, 通过对比可以了解各平台在用户体验上的差异。

• 合规性

在金融领域，合规性至关重要。分析各平台是否遵循相关金融法规和政策，如反洗钱、数据保护等，是评估其可靠性的关键。而且交易下单时是否会剧烈地影响市场波动也需要考虑在内。

• 生态建设与社区交流

包括平台能否提供良好的开发者社区、教育资源、合作伙伴网络等，这些都是评估平台长期发展潜力的重要指标。与券商或者高校合作，开发更多更优质的策略。

2.2 功能性需求

功能性需求编号	需求说明	优先级	备注
FR-01	用户注册与登录	高	数据库持久化保存
FR-02	采集历史数据和实时数据	高	
FR-03	可视化数据	高	
FR-04	开发用户界面	高	
FR-05	模拟交易	高	买入、卖出
FR-06	策略开发	高	
FR-07	多交易策略开发、组合	中	
FR-08	风险管理功能	中	自定义交易参数
FR-09	金融市场历史数据导出	中	
FR-10	历史交易记录查询	低	
FR-11	量化策略社区分享	低	

用户注册与登录

用户通过账号、密码登录，允许用户完善个人资料，如昵称、头像、性别、邮箱、简介等，支持修改密码。

数据采集和存储处理

从外部数据源采集实时和历史市场数据。对采集到的金融数据和用户数据利用数据库进行保存。支持将股票数据等金融数据导出，也支持用户查询他的历史交易记录。

策略开发和多策略组合

允许用户定义、修改和选择交易策略，对策略的自动化执行进行管理。允许用户对多个策略设定参数，进行组合。

模拟交易

根据策略生成交易信号，并管理买入、卖出等交易操作。可视化显示交易结果。对交易记录持久化保存下来。

2.3 设计约束

2.3.1 技术约束

高效性

在金融量化交易平台的设计中，高效性是至关重要的。平台需要能够在高并发环境下快速处理大量数据，同时还要适应资源受限的环境，如低内存或低带宽的网络环境。为此，平台采用高效的数据结构和算法以确保数据的快速访问和修改；利用多线程和异步编程技术，以及并发控制机制来支持多任务并发处理，提高在高并发环境下的处理能力；为了优化网络资源的利用，平台使用网络优化策略，如使用高效的通信协议和网络库，以及数据缓存和负载均衡策略。为了进一步优化性能，平台对代码进行优化，使用编译器优化选项和静态分析工具，生成高效的机器代码。通过这些方法，金融量化交易平台能够确保在高并发环境下快速处理大量数据，同时适应资源受限的环境，提高用户体验，降低硬件和运营成本。

兼容性

金融量化交易平台对兼容性有较高要求，以确保平台能够在不同的操作系统和设备上稳定运行，并且能够与其他系统和服务无缝集成。平台采用了多种策略和技术。首先，平台使用跨平台开发框架，实现代码在不同操作系统上的编译和运行。其次，平台遵循标准的编程实践和语言特性，使用标准的数据格式和通信协议，充分考虑与各种外部 API 和服务的兼容性（如交易所的 API、数据库管理系统等），确保代码的可移植性。此外，平台将针对不同的操作系统，调整软件的配置和资源管理策略，确保在 Windows、macOS 等系统上都能高效运行。最后，平台还将进行广泛的兼容性测试，确保软件在各种配置下都能正常工作。通过这些措施，金融量化交易平台能够确保跨平台兼容性，实现在不同操作系统和设备上的无缝运行，并且能够与其他系统和服务有效集成，提高平台的可用性和市场竞争力。

可扩展性

金融量化交易平台需要具有良好的可扩展性，能够适应未来业务增长和功能扩展的需求，为用户提供更加灵活和高效的服务。平台采用模块化设计，将系统划分为独立的模块，每个模块负责一个特定的功能。模块之间通过明确定义的接口进行通信，可以单独升级或

替换模块，而不影响其他部分。此外，平台将充分利用云计算服务提供的弹性计算能力，可根据需求扩展或缩小资源。

2.3.2 业务约束

合规性

金融量化交易平台在合规性方面必须遵守严格的金融法规和标准，以确保平台的合法运作和防范金融犯罪，如 AML（反洗钱）等。平台必须具备识别和报告可疑交易的能力，监测交易活动以识别和防止市场操纵和欺诈行为。同时，平台需要定期更新软件以符合最新的监管要求和法规变化，从而避免法律风险和潜在的财务损失，增强客户对平台的信任，维护公司在金融市场中的良好声誉。

安全性

金融量化平台应实施**客户身份识别（KYC）流程**，确保平台的用户身份真实可靠。平台需遵守数据保护和隐私方面的法律法规，如 GDPR（欧盟通用数据保护条例），确保客户数据的隐私和安全。

交易透明

金融量化平台需维护交易的透明度，确保所有交易活动都被适当记录和审计。根据法规要求，平台须**保留交易记录和客户通信记录**，并在规定的时间内提供给监管机构。

2.3.3 用户约束

界面简洁

金融量化平台界面须简洁直观，旨在提供高效、易用的用户体验。界面设计遵循清晰的布局和导航原则，确保用户能够快速找到所需功能。

高效交互

本平台将优化表单和交易流程，减少用户的操作步骤和时间。允许用户在交易执行前或执行中修改或取消交易，提供明确的指引和限制条件；同时，在用户尝试修改或取消交易时提供实时反馈，如交易执行状态、错误提示等，以满足用户的控制需求。

帮助文档

为了确保金融量化平台的用户和开发者能够高效地集成和使用平台，平台需要提供详细的 API 文档和系统接口规范。这些文档将涵盖 API 的详细说明、参数说明、使用示例以及错误处理指南，帮助开发者快速理解和使用平台提供的功能。

多语言

平台可能需要支持多语言，以适应不同国家和地区的用户。

2.4 依赖分析

对于我们开发的量化开发平台来说，包括数据分析库、交易执行引擎、数据源接口、第三方服务提供商等。

外部数据接口

1. 量化平台可能需要依赖外部数据提供商的实时和历史市场数据。这要求分析数据接口的稳定性、数据质量、访问速度和成本。
2. API 服务的依赖性也需要评估，包括它们的版本更新频率、文档完备性、技术支持和社区活跃度。

常用框架

我们需要选取一些框架便捷化我们的开发。

1. 数学库 (NumPy、SciPy)
2. 数据分析库 (Pandas)
3. web 应用框架 (Flask)
4. 前端框架 (Vue.js)

算力资源

- **云服务器：**我们将评估不同提供商的服务水平协议 (SLA)、成本和可扩展性。
- **硬件：**我们也将解决计算资源的依赖性

相关插件

1. 量化平台可能集成第三方 SDK 来实现特定的功能，如身份验证、支付处理或社交分享。
2. 插件系统的设计需要考虑插件的兼容性、安全性和更新管理。

开源项目

使用开源项目可以加速开发进程，但同时也带来了对项目维护、更新和社区活跃度的依赖。需要评估开源许可证的兼容性以及潜在的版权和专利问题。

- 兼容性：依赖项是否与当前系统的其他部分兼容，包括操作系统、编程语言版本等。
- 稳定性：依赖项的稳定性和成熟度，以及是否有足够的文档和社区支持。
- 安全性：依赖项的安全性记录，包括是否有已知的安全漏洞和及时的修复更新。
- 性能：依赖项对系统性能的影响，包括执行速度、资源消耗和网络延迟。
- 成本：依赖项的许可证成本、订阅费用或按使用量付费的模型。
- 维护和更新：依赖项的维护状态，是否有定期更新和长期维护的承诺。

2.5 影响分析

2.5.1 公共框架、控件相关软件影响分析

1. **兼容性**：本量化交易平台能与不同版本的公共框架、控件和模块兼容。本量化交易平台依赖于多个公共框架，如 UI 框架、数据库获取框架、量化交易回测框架。本平台以 *Python* 作为后端的开发语言，调用 *AlphaVantage* 和 *QuantConnect* 提供的 API 来获取数据、执行交易等操作，这与 *Alpha Vantage* 和 *QuantConnect* 所兼容。而利用 *React* 作为前端开发时则使用 *JavaScript* 语言。在 *Python* 后端中编写 *RESTfulAPI*，通过 *HTTP* 请求和响应与前端进行通信。前端使用 *JavaScript* 调用这些 API 来获取数据和执行操作，从而实现前后端的兼容性。。
2. **性能**：公共框架和控件的性能直接影响到平台的性能。本平台优先选择高效稳定的框架和控件，如 *React* 框架、*Bootstrap* 作为 UI 库，以提升交易平台的响应速度和处理能力，保障用户体验。
3. **可维护性**：本平台选择的成熟公共框架和控件可以提高软件的可维护性。本平台使用的 *Alpha Vantage* 和 *QuantConnect* 等均有完善的文档支持和社区维护，便于问题的定位和解决。

2.5.2 对用户的影响分析

1. 本平台向用户提供金融市场历史数据和实时数据，允许用户导出历史数据，满足用户对历史数据深入分析的需求，提供更多的研究可能性，帮助用户做出更好的交易决策。
2. 通过可视化数据和优化用户界面，可以显著提升用户体验满意度。直观、清晰的数据展示和操作界面能够使用户更轻松的理解和分析市场情况，提高用户操作的直观性和便捷性，进而增强用户对平台的信任和忠诚度。

3. 策略开发、多策略组合功能为用户提供了多样化的交易选择和风险管理工具，用户可以利用这些功能开发和优化自己的交易策略，根据不同的市场情况和个人偏好进行灵活调整，从而增强了用户的交易自主性和参与度，提高了用户的满意度和粘性。
4. 历史交易记录查询、量化策略社区分享功能，能够帮助用户回顾和分析自己的交易行为，并且为用户提供了交流与合作的平台，激发新的策略思路和创新。

3 系统方案设计

3.1 零层设计

本量化模拟交易平台系统涵盖了三个主要模块：数据获取与存储模块、用户服务模块以及股票模块。

本系统涉及到的周边交互模块包括 *Alpha Vantage* 数据源和 *Quant Connect* 回测框架。（由于使用限制，最终使用了爬虫结合 *AKshare* 库数据源和 *PyBroker* 回测框架）

用户服务模块，覆盖了用户注册、登录、密码管理、注销账户以及修改个人信息等核心功能，同时还提供社区交流分享功能。用户服务模块中生成的用户数据存入数据获取和存储模块中。此外，用户服务模块还负责向用户提供前端界面，支持用户查看金融数据、开发策略以及模拟交易，向股票模块发送请求，由股票模块完成处理后返回回测结果。

数据获取与存储模块，包括了用户数据的获取与存储以及金融数据的获取与存储。此模块调用周边交互模块 *Alpha Vantage* 数据源提供的 *API* 得到金融数据。为股票模块的策略开发提供数据支持。

股票模块，对于由用户服务模块发送的策略开发、因子回测、模拟交易的请求，利用周边交互模块 *Quant Connect* 回测框架提供的本地 *SDK* 完成，将回测结果返回到用户服务模块进行展示。并将交易记录传递到数据获取与存储模块进行保存。

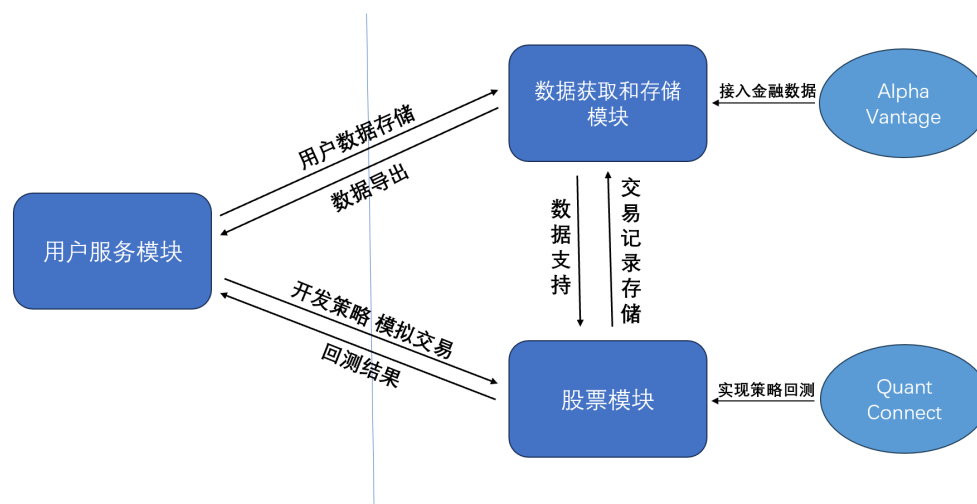


图 1: 零层设计示意图

3.2 一层设计

3.2.1 方案选型

- 系统架构

1. 使用 *Flask* 框架作为后端框架，处理 *HTTP* 请求和响应。
2. 前端可以采用 *HTML*、*CSS* 和 *JavaScript* 技术，使用 *Flask* 提供的 *API* 与后端通信。

- 数据存储

1. 使用 *MySQL* 作为后端数据库存储交易数据、用户信息等。
2. 可以设计多张表来存储不同类型的数据，例如交易记录表、用户信息表、资产表等。

- 后端逻辑

1. 设计 *Flask* 应用的路由和视图函数，处理用户请求并与数据库交互。
2. 实现用户认证和授权机制，确保用户的交易数据安全。
3. 编写交易策略逻辑，根据历史数据和用户输入执行交易策略，并将交易结果存储到数据库中。

- 前端展示
 1. 使用 *HTML*、*CSS* 和 *JavaScript* 构建交易系统的前端界面，提供用户交互功能。
 2. 可以使用前端框架（如 *Vue.js*、*React* 等）简化前端开发，提高用户体验。
 3. 展示交易数据、图表和交易结果等信息，让用户能够清晰地了解交易情况和结果。

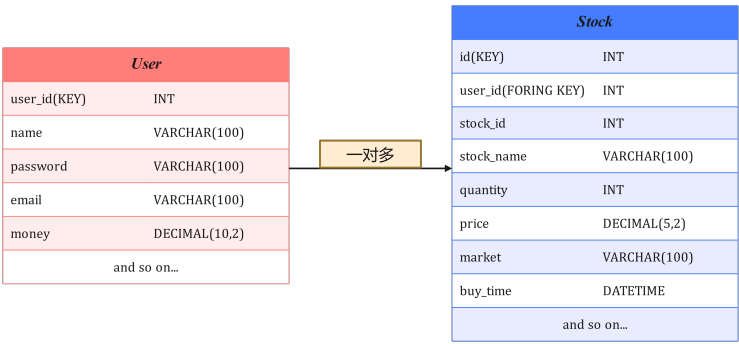


图 2: 数据库初步架构

3.2.2 整体框图

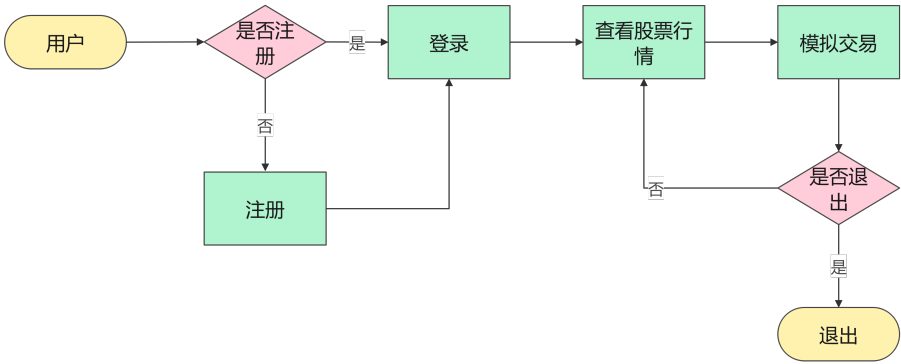


图 3: 整体框图

3.2.3 模块设计

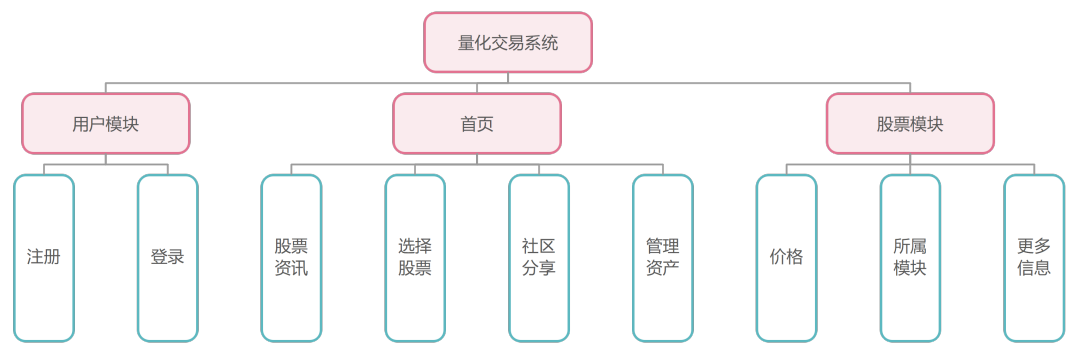


图 4: 模块设计

3.2.4 关键流程设计

具体的关键设计 PAD 图可见二层设计。

3.3 二层设计

用户注册与登录

1. 对应编号及描述

(a) 编号：FR-01

(b) 功能描述：本功能要求用户能够按照规定进行注册与登录。

关于注册，用户要输入邮箱进行注册，如果该邮箱没注册过则用户信息录入数据库，否则提示用户应该进入登录。邮箱是唯一识别用户的标识，可与用户形成一一映射，因为我们后面要发送验证码邮件，格式判断以是否能发送成功为标准。注册时在发送 5 分钟之内输入邮件验证码，超时要重新点击发送，用户输入正确之后，输入密码，用户需要包含数字和字母设定，而且不少于 8 位。登录需要用户输入邮箱和正确密码。如果邮箱存在且密码正确则进入；如果邮箱不存在则指引用户注册；如果用户存在但密码错误，支持连续尝试 5 次，超过次数则冻结 2 分钟。本产品支持用户进行忘记密码找回与修改，以及用户注销。密码找回需要用户输入正确的邮箱及发送的验证码，方可进行密码修改。密码修改需要用户输入正确的邮箱，原密码，验证码和符合格式要求的新密码，方可完成。注销需要用户输入正确的邮箱和密码，发送验证码并输入正确才可注销。

2. 功能设计

注册首先要判断邮箱是否符合格式以及注册过，然后判断密码是否符合要求，在判断验证码正确之后完成注册，信息录入数据库。

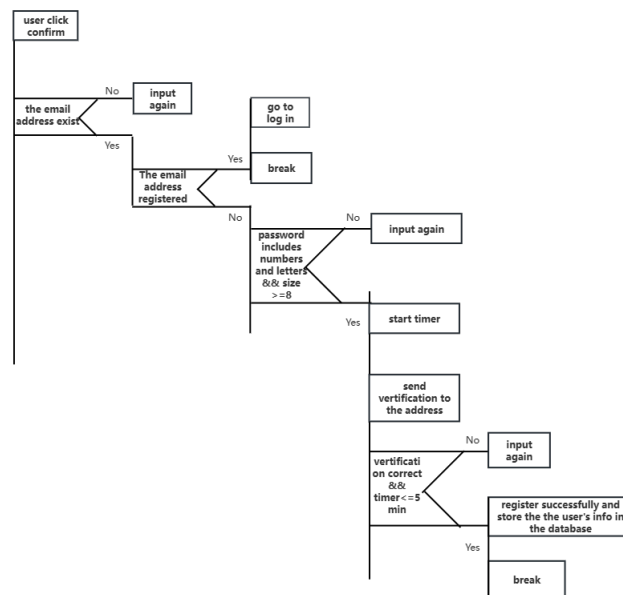


图 5: 用户注册 PAD 图

登录需要先判断邮箱是否存在，然后判断对应的密码是否正确，全正确即可完成。

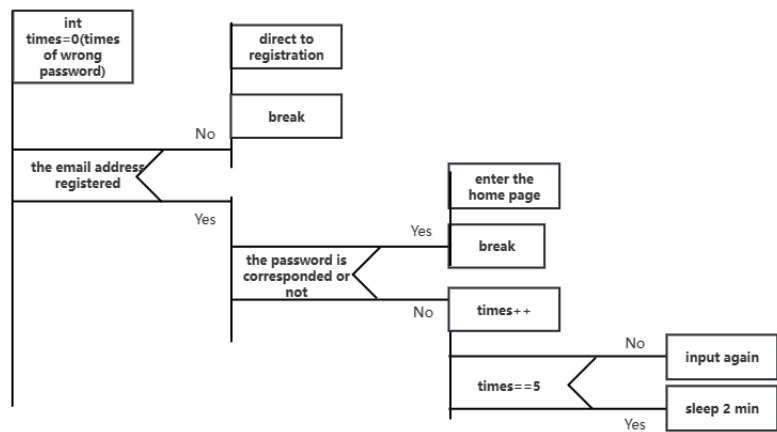


图 6: 用户登录 PAD 图

忘记密码修改需要先输入邮箱判断，然后输入验证码，正确后修改密码并判断格式是否符合要求，最后修改成功。

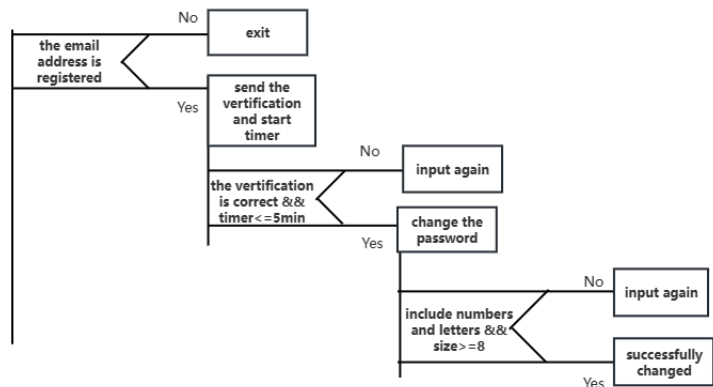


图 7: 用户忘记密码 PAD 图

修改密码需要先输入邮箱判断，然后输入对应密码，判断正确性，然后输入验证码，正确后修改密码并判断格式是否符合要求，最后修改成功。

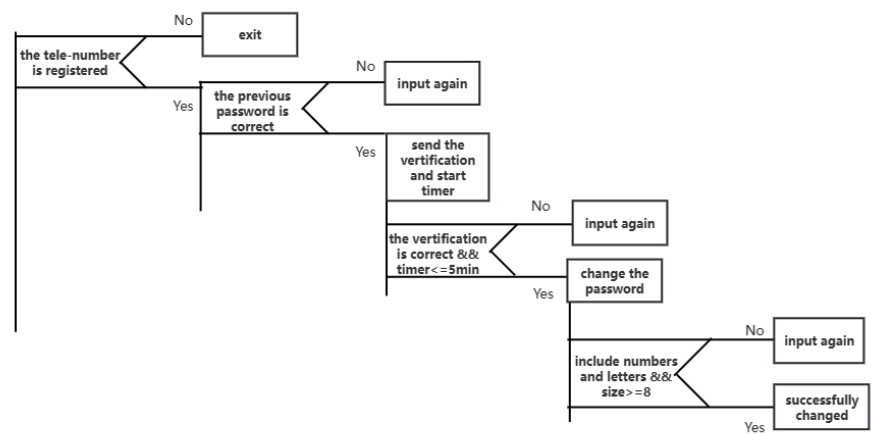


图 8: 用户修改密码 PAD 图

注销用户只要正确输入账号密码，以及正确的验证码即可。

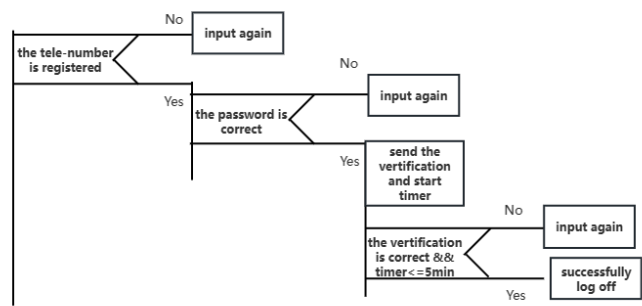


图 9: 用户注销 PAD 图

数据采集与可视化

1. 对应编号及描述

- (a) 编号：FR-02，FR-03
- (b) 功能描述：数据上，我们将调用支持实时市场数据的 API（暂定 *AlphaVantage*），根据其文档调用数据，并且进行可视化。主页上显示市场一览，此功能支持用户根据名称或者股票代码等搜索。支持子字符串搜索多结果展示。

2. 功能设计

当用户向输入框输入内容时，判断输入内容是否为某一结果的子串，给出选择列表，用户点击相应的按键，进入查看数据。

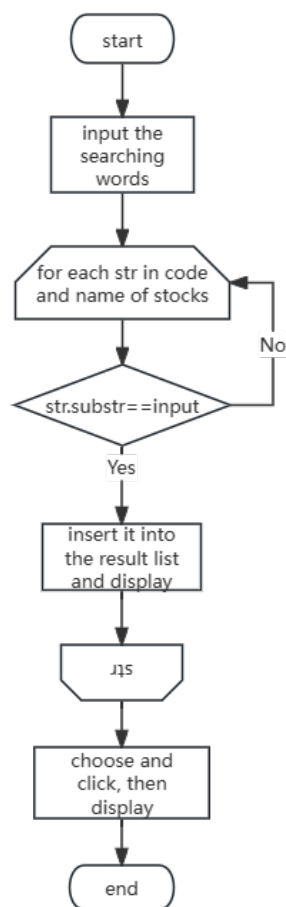


图 10: 搜索数据流程图

用户开发与模拟交易

1. 对应编号及描述

(a) 编号: FR-04, FR-05

(b) 功能描述: 本功能为用户开发策略及回测、实盘模拟交易等提供语言支持，本产品支持 Python 开发策略。

2. 功能设计

本功能在产品上搭载语言环境，允许用户在服务器上进行策略开发，自行编写策略运行回测和模拟实盘交易。

交易策略的开发、多种组合及风险管理

1. 对应编号及描述

(a) 编号：FR-06，FR-07，FR-08

(b) 功能描述：本功能支持内置策略，用户直接参考调用，并且支持参数修改进行风险管理。同时，本产品会根据策略的特点进行组合应用，提高安全性和收益性。

2. 功能设计

目前预计会加入比较经典的常用的策略，比如双均线策略，多因子选股策略等。对合适的策略进行量化多策略组合。即，通过投资相关性较低的股票、商品等资产，实现风险对冲，降低组合波动，提升稳健性。比如，指增策略与 CTA 策略的组合，将投资标的分散在相关性低的股市、商品市场，可分散风险、降低波动。

历史数据导出

1. 对应编号及描述

(a) 编号：FR-09

(b) 功能描述：本产品支持用户按照需求导出数据，文件格式支持.csv, .xlsx, .xls 等。用户本地可以处理数据。

2. 功能设计

在股票界面，设计都有导出数据的按键，用户点击导出数据，进入选择需求界面，选择所需的日期范围，再选择格式和文件路径，最后导出。

历史交易查询

1. 对应编号及描述

(a) 编号：FR-10

(b) 功能描述：本产品支持用户查询历史的买卖交易记录，便于用户梳理过程。支持用户根据购买的产品或日期查询，支持指定交易金额范围查询。

2. 功能设计

用户点击历史交易记录，直接进入查看。在顶部筛选中可以选择日期范围和金额范围，点击确认修改记录显示。

社区交流分享

1. 对应编号及描述

(a) 编号：FR-11

(b) 功能描述：本功能支持用户在社区发表策略研究，市场行情分享交流以及开发交流等。支持点赞评论，显示按照推荐或者发表时间排序。分区包括交流、问答、策略等主题。支持 markdown 格式发表。

2. 功能设计

用户在社区中可以进行编写、发表文章，还可以进行点赞、评论。文章内容显示支持按点赞数或者发表时间排序。编辑文章内容时，支持自动保存，支持插入图片、表情文字等。用户点击保存可以暂时保存草稿，点击发表时由用户先选择分区，然后发表。

阅读文章时，顶部加入筛选栏，可以按照按点赞数或者发表时间排序。

3.4 技术方案风险点

• 数据安全和隐私

金融量化交易平台涉及大量敏感数据，如交易记录、用户信息等，面临数据泄露和黑客攻击的风险。

解决方案

1. 实施严格的身份验证和权限管理，限制用户对敏感数据的访问。
2. 建立日志记录和审计机制，监控数据访问和操作，及时发现异常行为。

• 系统可靠性

交易平台需要保证高可用性和稳定性，避免系统崩溃或交易延迟导致的损失。

解决方案

定期备份数据和系统配置，确保在故障时能够快速恢复。

• 监管合规

金融领域的监管要求严格，交易平台需要遵守各项法规和规定，包括数据保护、交易透明度等方面，建立合规流程和监控机制至关重要。

解决方案

- 1. 建立符合监管要求的合规流程和操作规范，确保交易和数据处理符合法规。
- 2. 向用户提供合规培训和信息披露，增强用户对平台合规性的信任感，同时警示用户避免违规交易行为。
- 3. 定期邀请第三方机构对系统进行审计和合规检查，确保符合监管要求；聘请专业的法律顾问团队，及时了解和遵守金融行业的法规和政策变化。

• 算法模型风险

量化交易策略的成功依赖于算法模型的准确性和稳定性，可能存在过拟合、市场变化等风险。

解决方案

- 1. 定期对算法模型进行评估和验证，发现并修正可能的过拟合或失效问题。
- 2. 多样化策略：采用多种量化交易策略，分散风险，降低单一策略失效的影响。
- 3. 使用止损、资金管理等风险控制技术，实时监测市场数据和交易执行情况，及时调整策略和止损线，限制损失并保护投资资金。
- 4. 对历史数据进行充分的回测和验证，评估模型在不同市场环境下的表现，提高预测准确性。

4 数据结构设计

4.1 主要数据结构

主要的数据结构包括 User（用户）、Stock（股票）、TransactionRecord（交易记录）。展示如下：

User	
email (key)	VARCHAR(50)
password	VARCHAR(50)
user_name	VARCHAR(50)
user_id	VARCHAR(50)
gender	INT
avatar	VARCHAR(200)
retrv_password	# 找回密码
rev_password	# 修改密码
rev_name	# 修改昵称
rev_gender	# 修改性别
rev_avatar	# 修改头像
add_user	# 新增用户
del_user	# 注销用户

图 11: User 数据结构

User 数据结构包含关于用户的个人信息和关于用户的操作。将用户注册的 email 作为唯一标识，在用户注册成功后系统赋予用户唯一的编号 (user_id)，此外包含密码、昵称、性别、头像。

Stock	
stock_id (key)	VARCHAR(50)
stock_name	VARCHAR(100)

图 12: Stock 数据结构

Stock 数据结构只存储关于股票的市场编号和股票名称。其中编号是股票本身在市场中的唯一标识，也作为本数据结构的唯一标识。

TransactionRecord	
tr_id (key)	VARCHAR(50)
user_id	VARCHAR(50)
stock_id	VARCHAR(50)
time	DATETIME
op	INT
inv_amount	INT
return	INT
add_tr	# 新建交易记录
del_tr	# 删除交易记录

图 13: TransactionRecord 数据结构

TransactionRecord 数据结构记录用户的交易。系统为用户的每一条交易赋予一个编号 (tr_id) 作为交易记录的唯一标识。此数据结构还包括执行这次交易的用户编号 (user_id)，被交易的股票编号 (stock_id)，交易时间 (time)，交易的具体操作 (op，即买入还是卖出)，交易额 (inv_amount) 和收益 (return)。

4.1.1 数据库设计

数据库设计如下：

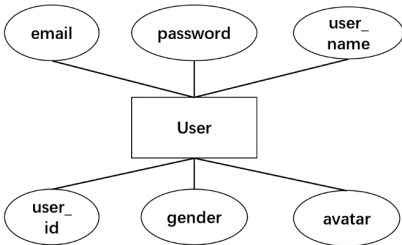


图 14: User 实体 E-R 图

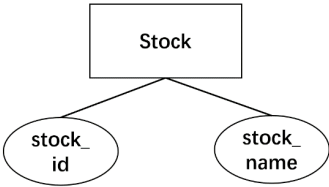


图 15: Stock 实体 E-R 图

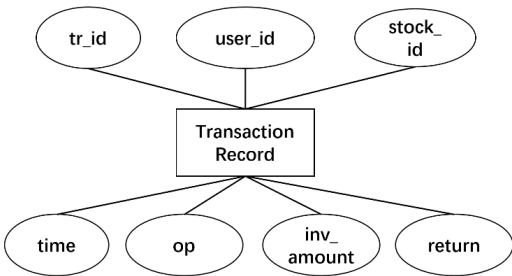


图 16: TransactionRecord 实体 E-R 图



图 17: 实体之间关系 E-R 图

一个用户可以拥有多个交易记录,通过在 TransactionRecord 数据库里查找列名为 user_id 的列,可以找到某个用户所有的交易记录。一条交易记录代表着对 stock_id 列对应的股票进行交易。

5 接口设计

5.1 提供接口

表 5: 提供的接口

接口	参数说明	返回值	功能说明	备注
/time	无	{ "time": <Unix 时间戳> }	获取服务器的当前时间	
/token	user_name (字符串): 用户名 email(字符串): 邮箱 password(字符串): 密码	{ "access_token": "<JWT 访问令牌>" }	根据用户信息生成访问令牌	根据用户信息生成访问令牌支持 GET 和 POST 方法
/stockdata	num (字符串): 页数	[{ "f2": <最新价>, "f12": <股票代码>, "f14": <股票名称> }]	获取指定页数的股票数据	
/buystock	stock_code(字符串): 股票代码 stock_name(字符串): 股票名称 stock_price(数字): 股票价格 stock_quantity (数字): 购买数量	{ "msg": "successful" }	用户购买股票并记录交易信息	需要 JWT 访问令牌
/stockdata_K	stock_code (字符串): 股票代码	[{ "date": "<日期>", "open": <开盘价>, "high": <最高价>, "low": <最低价>, "close": <收盘价>, "volume": <成交量> }]	获取指定股票的 K 线数据	

续下页

续页

接口	参数说明	返回值	功能说明	备注
/sellstock	stock_code (字符串): 股票代码 stock_name (字符串): 股票名称 stock_price (数字): 股票价格 stock_quantity (数字): 卖出数量	{ "msg": "successful" }	用户卖出股票并记录交易信息	需要 JWT 访问令牌
/search_buy	无	[{ "stockcode": "< 股票代码 >", "stockname": "< 股票名称 >", "stockprice": < 股票价格 >, "stockquantity": < 购买数量 >, "buytime": < 购买时间 > }]	获取交易记录	需要 JWT 访问令牌
/search_newprice	无	[{ "stock_code": "< 股票代码 >," "stock_num": < 股票数量 >, "stock_name": "< 股票名称 >," "stock_price": < 最新价格 > }]	获取用户持有股票的最新价格	需要 JWT 访问令牌

续下页

续页

接口	参数说明	返回值	功能说明	备注
/strategy	strategy_name (字符串): 策略名称 initial_cash (数字): 初始资金 param (对象): 参数对象, 包含 stock_code (字符串): 股票代码, percent (数字): 比例, stop_loss_pct (数字): 止损比例, stop_profit_pct (数字): 止盈比例, start_date (字符串): 开始日期, end_date (字符串): 结束日期	{ "result": "< 策略运行结果 >" }	根据提供的参数运行指定的策略	

5.2 依赖接口

表 6: 依赖的接口

接口	参数说明	返回值	功能说明	备注
ak.stock_zh_a_hist	symbol (字符串): 股票代码 start_date(字符串): 起始日期 end_date(字符串): 结束日期	股票历史数据的 DataFrame	获取指定股票代码在指定日期范围内的历史数据	

续下页

续页

接口	参数说明	返回值	功能说明	备注
ak.stock_financial_analysis_indicator	code (字符串): 股票代码	财务指标数据的 DataFrame	获取指定股票代码的财务指标数据	用于莫伦卡选股方法
ak.stock_zh_a_spot	无	A 股上市公司的实时行情数据 DataFrame	获取 A 股上市公司的实时行情数据	用于莫伦卡选股方法
pb.StrategyConfig	initial_cash (数字): 初始资金	策略配置对象	创建策略配置	
strategy.add_execution	fn (函数): 策略函数 symbols (字符串): 股票代码	无	添加策略执行方法	
strategy.backtest	无	回测结果对象	执行策略回测	

6 系统测试

6.1 概述

本章描述了量化金融交易平台的测试计划，包括单元测试和系统测试。测试旨在确保平台的功能性、性能、可靠性和安全性。

6.2 测试范围

测试范围涵盖平台的所有关键模块，包括数据获取与处理、交易策略执行、风险管理、用户接口、和系统集成等。

6.3 单元测试 V1

6.3.1 V1.1 白盒测试

测试模块：注册、登录、股票查询
测试时间：2024 年 5 月 29 日前

测试人员：张戡欣、陈巧蕾、王得而、喻思文

测试描述：白盒测试主要由设计人员完成，针对系统的内部逻辑和代码实现进行详细测试，确保代码的正确性和功能的完整性。测试包括以下几个方面：

- 注册模块：
 1. 验证用户输入的合法性（如用户名、密码的格式）。
 2. 检查用户数据是否正确写入数据库。
 3. 确保异常情况（如两次密码输入不同）能正确处理并提示用户。
- 登录模块：
 1. 验证用户输入的用户名和密码是否正确。
 2. 检查前后端用户身份认证是否正确实现。
 3. 确保异常情况（如用户名或密码错误）能正确处理并提示用户。
- 股票查询模块：
 1. 检查 API 数据获取和处理的正确性。
 2. 检查股票筛选排序功能。
 3. 确认 K 线图的正常展示和更新。
 4. 确认翻页时股票数据的刷新。

测试结果：

- **注册模块：**未实现邮箱的查重功能，已修改上传。拟增加邮箱验证码功能。
- **登录模块：**所有测试用例均通过，代码逻辑正确，功能实现无误。
- **股票查询模块：**K 线图所需数据前后端传输格式有误，未能正常展示，已修改。

6.3.2 V1.2 黑盒测试

测试模块：注册、登录、股票查询

测试时间：2024 年 5 月 30 日

测试人员：张戡欣、陈巧蕾、王得而、喻思文

测试描述：黑盒单元测试由测试团队在系统的外部功能上进行，不关注内部实现。测试主要针对用户的使用场景，确保系统功能从用户角度看是正确的和易用的。进行黑盒测试时，主要使用等价类法构造测试集：

- 注册模块：

1. 输入合法的邮箱、用户名和密码，检查是否能成功注册
2. 输入非法格式的邮箱、用户名或密码，检查是否能正确提示错误信息。
3. 输入已存在的邮箱，检查是否能正确提示用户名已存在。
4. 输入未注册的邮箱，检查是否能成功注册。
5. 两次密码输入不同，检查是否能成功注册。
6. 两次密码输入相同，检查是否能成功注册。
7. 输入错误的验证码，检查是否能成功注册。
8. 输入正确的验证码，检查是否能成功注册。

- 登录模块：

1. 输入非法的邮箱、用户名或密码，检查是否能正确提示错误信息。
2. 输入合法的邮箱、用户名和密码，检查是否能成功登录。
3. 输入正确的密码，检查是否能成功登录。
4. 输入不匹配的用户名和密码，检查是否能正确提示登录失败。
5. 检查登录后的用户会话管理是否正常

- 股票查询模块：

1. 调试不同指标的升降序排列按钮，检查是否能正确排序并获取股票信息。
2. 点击 K 线图的“上涨”和“下跌”按钮，检查是否能正确筛选数据并展示。
3. 检查查询结果的响应时间和数据正确性。

测试结果：所有测试用例均通过，但操作上不够 User-friendly。存在用户身份泄露问题，目前通过向后端发送请求，获取用户的邮箱和密码供前端验证，存在信息安全隐患。拟增加优化：（1）为每只股票添加 K 线图展示按钮；（2）修改前后端通信时的身份确认机制，为每个用户每次登录生成一个独特的 token 用于通信时确认身份，并不将用户信息直接传给前端。

6.3.3 V1.3 白盒测试

测试模块：个人信息、每日新闻、风险偏好评估、关于我们

测试时间：2024 年 6 月 6 日

测试人员：张戥欣、陈巧蕾、王得而、喻思文

测试描述：白盒测试主要由设计人员完成，针对系统的内部逻辑和代码实现进行详细测试，确保代码的正确性和功能的完整性。测试包括以下几个方面：

- 个人信息模块：
 1. 验证用户信息的修改功能（如性别、手机号等格式）。
 2. 检查用户信息是否正确保存到数据库。
 3. 确保数据更新的及时性，重点测试余额的变动。
- 每日新闻模块：
 1. 验证新闻数据的获取和显示功能。
 2. 检查新闻内容的链接跳转是否正确。
- 风险偏好评估模块：
 1. 验证评估问卷的各项问题及选项是否正确显示。
 2. 检查用户提交的答案是否正确处理。
 3. 确保异常情况（如未完整作答）能正确处理并提示用户。
 4. 检查风险偏好结果的显示。
- 关于我们模块：
 1. 验证关于我们页面的内容显示是否正确。
 2. 检查页面的各项信息是否准确且无错漏。
 3. 检查页面的联系方式等各项链接设置正确。

测试结果：所有测试用例均通过，代码逻辑正确，功能实现无误。

6.3.4 V1.4 黑盒测试

测试模块：个人信息、每日新闻、风险偏好评估、关于我们

测试时间：2024 年 6 月 7 日

测试人员：张彧欣、陈巧蕾、王得而、喻思文

测试描述：黑盒单元测试由测试团队在系统的外部功能上进行，不关注内部实现。测试主要针对用户的使用场景，确保系统功能从用户角度看是正确的和易用的。测试包括以下几个方面：

- 个人信息模块：
 1. 验证用户信息的修改功能（如性别、手机号等格式）。
 2. 检查用户信息是否正确保存到数据库。
 3. 确保数据更新的及时性，重点测试余额的变动。
- 每日新闻模块：
 1. 验证新闻数据的获取和显示功能。
 2. 检查新闻内容的链接跳转是否正确。
- 风险偏好评估模块：
 1. 验证评估问卷的各项问题及选项是否正确显示。
 2. 在未完整作答的情况下提交问卷，检查是否提示用户需要完成全部问卷
 3. 在完整作答的情况下提交问卷，检查是否正确提交并输出结果。
 4. 输入极端偏好风险的问卷答案，查看系统输出的结果。
 5. 输入极端厌恶风险的问卷答案，查看系统输出的结果。
- 关于我们模块：
 1. 验证关于我们页面的内容显示是否正确。
 2. 检查页面的各项信息是否准确且无错漏。
 3. 检查页面的联系方式等各项链接设置正确。

测试结果：所有测试用例均通过，系统功能从用户角度看是正确的，操作简便且响应及时。

6.3.5 V1.5 白盒测试

测试模块：策略回测、股票交易、我的资产、交易记录

测试时间：2024 年 6 月 9 日前

测试人员：张彧欣、陈巧蕾、王得而、喻思文

测试描述：白盒测试主要由设计人员完成，针对系统的内部逻辑和代码实现进行详细测试，确保代码的正确性和功能的完整性。测试包括以下几个方面：

- 策略回测模块：
 1. 验证策略代码的正确性和逻辑性。

2. 检查回测数据的获取和处理是否正确。
 3. 确保回测结果的计算准确性（如收益率、最大回撤等）。
- 股票交易模块：
 1. 验证买入、卖出功能的实现。
 2. 检查交易指令的执行和反馈是否正确。
 3. 确保交易过程中账户余额和持仓的正确更新。
 - 我的资产模块：
 1. 验证资产信息的显示和更新功能。
 2. 检查账户总资产、持仓股票、实时股价等信息的准确性。
 3. 确保异常情况能正确处理并提示用户。
 - 交易记录模块：
 1. 验证交易记录的保存和显示功能。
 2. 检查交易记录的完整性和准确性（如时间、类型、数量、价格等）。
 3. 确保记录查询功能的正确性和响应速度。

测试结果：策略回测模块后端程序逻辑有误，前后端数据调用格式存在问题，已尽快投入修改。其他模块所有测试用例均通过，代码逻辑正确，功能实现无误。

6.3.6 V1.6 黑盒测试

测试模块：策略回测、股票交易、我的资产、交易记录

测试时间：2024 年 6 月 10 日

测试人员：张戡欣、陈巧蕾、王得而、喻思文

测试描述：黑盒单元测试由测试团队在系统的外部功能上进行，不关注内部实现。测试主要针对用户的使用场景，确保系统功能从用户角度看是正确的和易用的。测试包括以下几个方面：

- 策略回测模块：
 1. 在不同时间（股票价格存在不同）测试回测策略，检查回测执行是否正确。
 2. 验证回测结果的各项指标（如收益率、最大回撤）是否准确。

3. 检查回测过程的响应速度和异常处理（如无效策略）。
- 股票交易模块：
 1. 买入股票，检查交易是否成功执行。
 2. 卖出股票，检查交易是否成功执行。
 3. 验证交易后账户余额和持仓是否正确更新。
 4. 买入超过自身资金的股票，检查异常情况是否报错。
 5. 检查异常情况处理，如当前股票不可交易。
 - 我的资产模块：
 1. 查看账户总资产、可用资金、持仓市值等信息，检查是否正确显示。
 2. 进行买入和卖出以改变余额，验证资产信息的实时更新。
 3. 重新登录，检查资产信息是否存储正确。
 - 交易记录模块：
 1. 生成多条交易记录，检查记录的完整性和准确性。

测试结果：策略回测程序运行有几率出现问题，已投入修改。其他模块所有测试用例均通过，系统功能从用户角度看是正确的，操作简便且响应及时。

6.3.7 V1.7 黑盒测试

测试时间：2024 年 6 月 10 日

测试人员：张彧欣、陈巧蕾、王得而、喻思文

测试描述：黑盒单元测试由测试团队在系统的外部功能上进行，不关注内部实现。测试主要针对用户的使用场景，确保系统功能从用户角度看是正确的和易用的。测试包括以下几个方面：

- 策略回测模块：
 1. 在不同时间（股票价格存在不同）测试回测策略，检查回测执行是否正确。
 2. 验证回测结果的各项指标（如收益率、最大回撤）是否准确。
 3. 检查回测过程的响应速度和异常处理（如无效策略）。

测试结果：所有测试用例均通过，系统功能从用户角度看是正确的，操作简便且响应及时。

6.4 系统测试 V2

6.4.1 V2.1 黑盒测试

测试时间：2024 年 6 月 10 日

测试人员：张彧欣、陈巧蕾、王得而、喻思文

测试描述：最终系统测试是对整个系统进行全面、系统化的测试，确保所有模块和功能的整体协调性、稳定性和可靠性。测试包括以下几个方面：

- 功能测试
 1. **注册和登录模块：**验证用户从注册到登录的整个流程是否顺畅，信息是否准确保存和处理。
 2. **股票查询模块：**检查用户在查询股票信息时的操作是否正确，信息是否及时更新。
 3. **个人信息模块：**验证用户个人信息的查看、修改功能是否正常。
 4. **每日新闻模块：**检查每日新闻的显示和更新是否准确及时。
 5. **风险偏好评估模块：**验证用户进行风险偏好评估的过程和结果是否正确。
 6. **关于我们模块：**检查关于我们页面的内容显示是否准确无误。
 7. **策略回测模块：**验证不同策略回测的结果和数据处理是否正确。
 8. **股票交易模块：**检查用户进行股票交易的整个流程，确保买入卖出指令执行无误。
 9. **我的资产模块：**验证用户账户资产信息的显示和实时更新功能是否正常。
 10. **交易记录模块：**检查交易记录的保存和查询功能是否准确无误。
- 性能测试：
 1. **响应时间：**测试系统各模块在高负载下的响应时间，确保系统在大流量下仍能快速响应。
 2. **负载测试：**模拟大量用户同时操作，测试系统的稳定性和承受能力。
- 安全测试：
 1. **用户数据保护：**验证用户数据的存储和传输是否加密，确保数据安全。
 2. **权限控制：**检查系统的权限控制机制，确保不同角色的用户只能访问其权限范围内的功能和数据。
- 兼容性测试：

1. **浏览器兼容性：**测试系统在不同浏览器（如 Chrome, Firefox, Safari, Edge）上的表现，确保页面显示和功能操作一致。
 2. **设备兼容性：**验证系统在不同设备（如 PC, 平板, 手机）上的兼容性，确保跨设备使用无障碍。
- **用户体验测试：**
 1. **界面友好性：**评估系统界面的美观度和易用性，确保用户操作简便、体验良好。
 2. **交互流畅性：**检查系统的交互设计是否合理，操作流程是否顺畅无阻。

测试结果：所有测试项目均通过，系统功能、性能、安全性和兼容性都达到了预期标准，用户体验良好。优化系统界面，调整了几处不统一的图标和大小不适宜的窗口，并再次测试。系统整体运行稳定可靠，满足上线要求。

6.4.2 V2.2 黑盒测试

测试时间：2024 年 6 月 11 日

测试人员：2 位未参与开发的同学（用户）

测试描述：为确保系统在实际使用中的效果和用户体验，我们邀请了 2 位未参与开发的同学进行系统测试。他们从用户的角度对系统的各个模块进行了全面的测试，主要测试项目包括：

- **功能测试**
 1. **注册和登录模块：**用户注册新账户并登录，检查注册和登录流程的便捷性和正确性。
 2. **股票查询模块：**用户查询股票信息，检查信息的准确性和实时性。
 3. **个人信息模块：**用户查看和修改个人信息，验证信息的正确显示和修改后的保存。
 4. **每日新闻模块：**用户浏览每日新闻，检查新闻内容的更新和显示是否正常。
 5. **风险偏好评估模块：**用户进行风险偏好评估，验证评估过程的流畅性和结果的准确性。
 6. **关于我们模块：**用户浏览关于我们页面，检查内容的准确性和页面设计的美观度。
 7. **策略回测模块：**用户设置不同策略进行回测，验证回测结果的准确性和界面的友好度。
 8. **股票交易模块：**用户进行股票买入和卖出操作，检查交易过程的顺畅性和交易结果的正确性。

9. **我的资产模块：**用户查看账户总资产、可用资金、持仓市值等信息，检查信息的正确显示和实时更新。

10. **交易记录模块：**用户查看交易记录，验证记录的完整性和准确性。

- **用户体验测试：**

- 1. **界面友好性：**评估系统界面的美观度和易用性，确保用户操作简便、体验良好。

- 2. **交互流畅性：**检查系统的交互设计是否合理，操作流程是否顺畅无阻。

测试结果：用户测试结果显示，系统功能实现正确，操作流畅，界面友好。用户在使用过程中未发现明显的功能性问题，整体体验良好。所有模块均通过了用户测试。