详细设计

——基金优选APP

目录

[详细设计 1](#_Toc85666960)

[——基金优选APP 1](#_Toc85666961)

[1. 概述 3](#_Toc85666962)

[1.1 系统简述 3](#_Toc85666963)

[1.2 软件设计目标 3](#_Toc85666964)

[1.3 修订版本记录 4](#_Toc85666965)

[2. 术语表 6](#_Toc85666982)

[3. 设计概述 7](#_Toc85666983)

[3.1 系统的复用计划 7](#_Toc85666984)

[3.2 系统接口设计 8](#_Toc85666985)

[3.3 对象模型设计 8](#_Toc85666986)

[3.3.1对象模型 8](#_Toc85666987)

[3.3.2对象描述 8](#_Toc85666988)

[3.4 系统用例实现详细设计 14](#_Toc85666989)

[3.4.1总用例图 14](#_Toc85666990)

[3.4.2用例描述 14](#_Toc85666991)

[3.5 系统非功能设计 18](#_Toc85666992)

[3.5.0 约束矩阵 18](#_Toc85666993)

[3.5.1 性能设计 19](#_Toc85666994)

[3.5.2 维护性设计 19](#_Toc85666995)

[3.5.3 易用性设计 19](#_Toc85666996)

[3.5.4 兼容性设计 19](#_Toc85666997)

[3.5.5 可拓展性设计 20](#_Toc85666998)

[3.5.6 可靠性设计 20](#_Toc85666999)

[3.5.7 灾备设计 20](#_Toc85667000)

[3.5.8 安全性设计 20](#_Toc85667001)

# 概述

## 1.1 系统简述

基金优选APP是一款能够给我们提供基金投资便利的手机软件，其迎合市场需求，能够为基金投资者（以下简称基民）提供简单、低门槛的基金浏览、检索服务；能够让基民快捷地记录心仪的基金；能够让基民方便地查看基金信息。

相比于天天基金网之类的股票基金交易的综合性平台，基金优选APP的主要功能是作为一个发掘用户期望基金的推荐平台

在复写传统平台上的持仓搜索功能的同时，我们发现了传统平台在板块搜索方面的缺陷，针对次痛点我们综合考虑心仪版块数量，重仓股占比以及单基金的期望板块集中度，实现了更贴合用户的板块搜索及其衍生方法，并完善相关功能。

## 1.2 软件设计目标

实现一个包括账户模块账户模块、搜索模块、用户管理基金模块、股票和基金管理模块、系统管理模块、统计模块等六大模块的APP。其中账户模块是许多IT系统的基本模块，用户可以注册、登录账户，并编辑修改账户的用户名、邮箱等信息；搜索模块是基金优选APP的核心模块之一，用户可以进行基金和股票搜索，也可以根据股票或者板块进行基金的筛选；用户管理基金模块是基金优选APP的核心模块之一，用户可以添加心仪的的基金，或删除不再关注的基金；股票和基金管理模块是基金优选APP的核心模块之一，将股票或基金的详细分析上推到系统中，呈现在用户可浏览视图中；系统管理模块基金优选APP的核心模块之一，管理员通过管理界面查看用户信息、业务员信息、商品信息、业务员/统计人员授权、用户封号等操作，以及开启和关闭系统；统计模块基金优选APP的核心模块之一，统计人员根据系统的业务数据进行相关的用户和基金选择的统计，为商业决策提供建议。

|  |
| --- |
| 经济学基金基础架构备忘 |
|  |
| 项目中的用户需求架构备忘 |
| 持仓搜索与之类似 |

在非功能性需求方面，系统响应速度：用户操作系统平均响应时间小于3秒。支持最大并发数为50。在维护方面整个应用软件系统应能够连续7\*24小时不间断工作，应用软件中的任一模块更新、加载时，在不更新与上下模块的接口的前提下，以不影响业务运转和服务为原则；应用软件具有较高的自动化程度，如：自动异常调度、自动故障告警、自动任务恢复等；应用软件具备相应容错手段，能容许操作人员的某些失误操作。在安全性上包括用户和权限管理，数据范围权限的控制，对支付通讯进行加密，防止数据篡改、攻击。

## 1.3 修订版本记录

|  |  |  |  |
| --- | --- | --- | --- |
| **版 本** | **日 期** | **描 述** | **作 者** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046)**[#2](https://github.com/Dawn1261283562/Android-foundation/pull/5)**[from Dawn1261283562/LiaoPenglin](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046) | Commits on Sep 9, 2021 | Commits on Sep 9, 2021 user类代码更新 | **Dawn1261283562** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046)**[#5](https://github.com/Dawn1261283562/Android-foundation/pull/5)**[from Dawn1261283562/LiaoPenglin](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046) | Commits on Sep 15, 2021 | Commits on Sep 15, 2021批量搜索 | **Dawn1261283562** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046)**[#7](https://github.com/Dawn1261283562/Android-foundation/pull/5)**[from Dawn1261283562/LiaoPenglin](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046) | Commits on Sep 24, 2021 | Commits on Sep 24, 2021登录 ，相关文档 | **Dawn1261283562** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046)**[#10](https://github.com/Dawn1261283562/Android-foundation/pull/5)**[from Dawn1261283562/LiaoPenglin](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046) | Commits on Sep 27, 2021 | Commits on Sep 27, 2021模式三 数据库简化 | **Dawn1261283562** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/2464150027e1a9d4c73b87db8c4332791a0deba2)**[#13](https://github.com/Dawn1261283562/Android-foundation/pull/4)**[from Dawn1261283562/CaiChengjie](https://github.com/Dawn1261283562/Android-foundation/commit/2464150027e1a9d4c73b87db8c4332791a0deba2) | Commits on Sep 28, 2021 | Commits on Sep 28, 2021 bug | **[ccj198754](https://github.com/Dawn1261283562/Android-foundation/commits?author=ccj198754" \o "View all commits by ccj198754)** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046)**[#16](https://github.com/Dawn1261283562/Android-foundation/pull/5)**[from Dawn1261283562/LiaoPenglin](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046) | Commits on Oct 13, 2021 | Commits on Oct 13, 2021 bug 修复 | **Dawn1261283562** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046)**[#21](https://github.com/Dawn1261283562/Android-foundation/pull/5)**[from Dawn1261283562/LiaoPenglin](https://github.com/Dawn1261283562/Android-foundation/commit/0bfb22e96b7981b08ff548b1224b7e8b88764046) | Commits on Oct 18, 2021 | Commits on Oct 18, 2021增加股票热点 | **Dawn1261283562** |
| [Merge pull request](https://github.com/Dawn1261283562/Android-foundation/commit/0d666dcd151e889232463b2f584d613e3e7ace13)**[#26](https://github.com/Dawn1261283562/Android-foundation/pull/26)**[from Dawn1261283562/LinJie](https://github.com/Dawn1261283562/Android-foundation/commit/0d666dcd151e889232463b2f584d613e3e7ace13) | Commits on Oct 20, 2021 | Commits on Oct 20, 2021 股票动态更新 | **[Zeohhhh](https://github.com/Dawn1261283562/Android-foundation/commits?author=Zeohhhh" \o "View all commits by Zeohhhh)** |

# 术语表

|  |  |  |
| --- | --- | --- |
| 编号 | 术语 | 定义和信息 |
| 1 | 大类行业 | 股票所属于的行业，可能有多个。 |
| 2 | 板块 | 板块是指某些公司在股票市场上有某些特定的相关要素，就以这一要素命名该板块。板块一般分为三大类：行业板块、区域板块、概念板块。每个大类又分成若干板块.此外还有指数板块和上市时间分类的板块。 |
| 3 | 板块搜索 | 系统根据用户输入的板块信息来给出基金或股票信息。 |
| 4 | 持仓 | 股票持仓就是继续持有股票。 意思就是你用你帐户里的钱买了股票，然后你买的股票一直在你的帐户里放着，没有把你买的这些股票卖掉，这种状态叫做持仓。 |
| 5 | 持仓搜索 | 指通过搜索股票，系统给出该股票持仓占比为前十的基金 |
| 6 | 持仓占比 | 持仓占比就是当前持有的股票占有总资产的比例是多少，仓位占比=买入金额/总金额\*100%。在本项目中指基金中股票的金额占总金额的百分比 |
| 7 | 单位净值 | 单位净值是基金单**位份额的价格。 单位**净值的计算方法是，将组合中股票、债券或其他有价证券 (包括现金)的价值加总，并扣除相关的费用，将余额除以总份额。 |
| 8 | 概念板块 | 概念板块： 锂电池 军工航天 新能源 低碳经济 稀缺资源 循环经济 物联网 智能电网 铁路基建 黄金概念 3G概念 股权激励 基金重仓 资产注入 分拆上市 稀土永磁 多晶硅。 |
| 9 | 股票 | 股份公司发行的所有权凭证。股东对公司出资，获得了相应的股份和股权，股票便是由公司签发的权利凭证。凭借股票，股东可以分享到相应的公司利润，但是同时也要承担公司会遇到的各种风险。 |
| 10 | 基金 | 广义是指为了某种目的而设立的具有一定数量的资金。本项目中专指用于投资多只有潜力股票的资金。 |
| 11 | 基金净值 | 基金净值可以分为单位净值和累计净值。 |
| 12 | 季涨幅 | 近3月的涨幅 |
| 13 | 净值估算 | 净值估算是指通过往年的净值曲线预估未来的净值走势。 |
| 14 | 累计净值 | 累计净值是基金单位净值与基金成立后历次累计单位派息金额的总和，反映该基金自成立以来的所有收益的数据。基金单位累计净值=基金单位净值+基金历史上累计单位派息金额(基金历史上所有分红派息的总额/基金总份额）。 |
| 15 | 区域板块 | 区域板块： 黑龙江省 新疆板块 吉林板块 甘肃板块 辽宁板块 青海板块 青海板块 北京板块 陕西板块 天津板块 广西板块 河北板块等 |
| 16 | 行业板块 | 金融行业 首饰行业 钢铁行业 家具行业 石油行业 公路桥梁 汽车行业 交通运输 医疗器械 酒店旅游 房地产业 商业百货 物资外贸等. |
| 17 | 涨幅 | 基金净值的变化值占变化前的净值的百分比 |
| 18 | 重仓 | 重仓就是把自己大部分的资金买了多支股票，其中一只投资规模比例比其他的股票大，那么这支比例大的股票就是重仓持有票。 |
| 19 | 重仓股 | 重仓股是指股票被多家基金公司、某一机构或大户大量买进并持有,买入的资金占股票总流通市值大,这类股票则为重仓股。本项目中指在基金中持仓占比前十的股票 |

# 设计概述

## 3.1 系统的复用计划

我们首先进行基金优选的移动前端开发，我们计划未来将进行其web前端开发，此时前端变成外部界面，我们将更换前端架构，但仍使用原来系统的后端以及数据库；基金优选APP目前主要通过本地数据库来维护，我们计划未来将改变数据来源，数据请求和数据更新将更改为通过后端使用API来完成，仍使用原来系统的前端以及数据库。

其中，我们复用了后端的Spring架构，我们在Spring中使用工厂模式。工厂模式是一个创建对象的接口，不同的子类可以创建不同的对象。我们将在需要创建复杂对象的时候使用工厂模式，这使得我们的代码结构更加清晰，能够有效地封装变化。

我们还复用了以下但不限的相关库：

（1）import org.springframework.web.client.RestTemplate

（2）import org.springframework.jdbc.core.JdbcTemplate

（3）import org.springframework.jdbc.core.RowMapper

（4）……

本系统中没有进行框架和构件的复用（后端主体Spring框架不变），软件过程中也未定义复用设计师、复用工程师的角色；项目规模较小，无复用需求（但后面会使用相关比赛的数据API）；功能模块较为独立，无需复用构件支持。

## 3.2 系统接口设计

### 3.2.1内部接口

1. 登录接口
2. 请求方式:POST
3. 请求地址：[http://4s348z6897.qicp.vip/loginServlet](http://localhost:8080/mall/index.html)
4. 请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| username | String | 是 | 用户名 |
| password | String | 是 | 密码 |

1. 返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| user | User | 用户 |

1. 后端接口
2. UserDao：定义了对用户信息进行增删改查的四个函数接口。

*/\*\*增\*\*/*int insert(User user);  
*/\*\*删\*\*/*int deleteByPhone(String username);  
*/\*\*改\*\*/*int update(User user);  
*/\*\*查\*\*/*User getByPhone(String username);  
List<User> getListByPhone(String username);

1. StockDao：定义了对股票信息进行查找和更新，以及更新股票热度的函数接口。

*/\*\*根据ID查找股票\*\*/*Stock getById(String id);  
*/\*\*更新股票热度\*\*/*public int updateHitsBySearch(String id) ;  
*/\*\*更新股票信息\*\*/*public int update(Stock stock) ;

1. FundHeavyDao：定义了对重仓股的操作

*/\*\*获取重仓股信息\*\*/*List<FundHeavy> getListAll();  
*/\*\*加载重仓股\*\*/*void setFundHeavyAll();  
*/\*\*预操作\*\*/*void preprocess();

### 3.2.2外部接口（暂无）

## 3.3 对象模型设计

### 3.3.1对象模型

后端类图（主要部分，用户收藏映射在数据库中）：

|  |
| --- |
|  |

前端类图：

|  |
| --- |
|  |

### 3.3.2对象描述

**3.3.2.1对象属性描述**

|  |  |  |
| --- | --- | --- |
| 对象：FundHeavy | | |
| 属性名 | 类型 | 描述 |
| Score | Double | 通过一定的算法为每一个基金评分，以便于最后进行排序 |
| Id | String | 基金独有的ID，便于与其他基金区分 |
| Name | String | 基金独有的名称，便于与其他基金区分 |
| Stock\_id | String[ ] | 用于存储基金中股票的ID |
| Stock\_name | String[ ] | 用于存储基金中股票的名称 |
| Stock\_ratio | String[ ] | 股票在该基金中的所占比例 |
| Stock\_all\_type | String[ ] | 该基金中股票所属的类型 |

|  |  |  |
| --- | --- | --- |
| 对象：FundHeavyDao | | |
| 属性名 | 类型 | 描述 |
| jdbcTemplate | JdbcTemplate | 为多种关系数据库提供统一访问 |
| stockDaoImpl | StockDaoImpl | 使其能用StockDao层函数 |
| fundHeavyAll | List< > | 将数据库中所有的基金重仓信息存到后端 |
| fundHeavyAllTemp | List< > | 优化数据库访问，和第1个表拼接 |

|  |  |  |
| --- | --- | --- |
| 对象：FundHeavyController | | |
| 属性名 | 类型 | 描述 |
| fundHeavyService | FundHeavyService | 使其能用service层 |

|  |  |  |
| --- | --- | --- |
| 对象：FundHeavyService | | |
| 属性名 | 类型 | 描述 |
| fundHeavyDao | FundHeavyDao | 使其能用dao层 |

|  |  |  |
| --- | --- | --- |
| 对象：User | | |
| 属性名 | 类型 | 描述 |
| Id | Long | 用户ID，区分不同账户 |
| Username | String | 用户名，用于与其他账户进行区分 |
| Password | string | 密码，用户登录时使用，防止陌生人的盗用 |

|  |  |  |
| --- | --- | --- |
| 对象：UserDao | | |
| 属性名 | 类型 | 描述 |
| jdbcTemplate | JdbcTemplate | 为多种关系数据库提供统一访问 |

|  |  |  |
| --- | --- | --- |
| 对象：UserController | | |
| 属性名 | 类型 | 描述 |
| userService | UserService | 使其能用service层 |

|  |  |  |
| --- | --- | --- |
| 对象：UserService | | |
| 属性名 | 类型 | 描述 |
| userDao | UserDao | 使其能用dao层 |

|  |  |  |
| --- | --- | --- |
| 对象：Stock | | |
| 属性名 | 类型 | 描述 |
| Id | String | 股票特有的ID，用于区分不同的股票 |
| Name | String | 股票独有的用户名，用于区分不同的股票 |
| Type | String | 股票所属的类型 |
| Price | String | 股票的单股价格 |
| Price\_change | String | 股票价格变化 |
| Price\_radio | String | 股票价格变化比率 |

|  |  |  |
| --- | --- | --- |
| 对象：StockDao | | |
| 属性名 | 类型 | 描述 |
| jdbcTemplate | JdbcTemplate | 为多种关系数据库提供统一访问 |

|  |  |  |
| --- | --- | --- |
| 对象：StockController | | |
| 属性名 | 类型 | 描述 |
| httpClient | HttpClient | SpringCloud中服务和服务之间的调用 |
| stockService | StockService | 使其能用service层 |

|  |  |  |
| --- | --- | --- |
| 对象：StockService | | |
| 属性名 | 类型 | 描述 |
| stockDao | StockDao | 使其能用dao层 |

**3.3.2.2对象方法描述：**

方法名，返回类型，返回值，参数，用途以及使用的算法的简要说明：

基金重仓信息功能是项目核心需求功能，主要分为两大功能模块：持仓搜索与板块搜索，其中是投资网站的普适性需求，而 板块搜索

经济学基金基础架构备忘/项目中的用户需求架构备忘

|  |
| --- |
| 经济学基金基础架构备忘 |
|  |
| 项目中的用户需求架构备忘 |
| 持仓搜索与之类似 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 基金重仓信息检索与主要功能(service层)  @Service  public class FundHeavyServiceImpl implements FundHeavyService  基金重仓信息功能是项目核心需求功能，主要分为两大功能模块：持仓搜索与板块搜索，其中是基金投资网站的普适性需求，而 板块搜索 在目前现有网站过于单一，不能满足用户“寻找指定板块而非股票”的需求，是需主要解决的痛点之一。  下面是检索功能的部分简要说明，后面会有部分详细说明。 | | | | | | | |
|  |  | 返回类型 | 方法名 | | 参数 | 说明 | |
| 1 | 简要说明 | **//获取全部基金重仓持有信息** | | | | | |
| 函数申明 | List<FundHeavy> getListAll(); | | | | | |
| 2 | 简要说明 | **//模式一: 持仓搜索，基于持仓股种类** | | | | | |
| 函数申明 | List<FundHeavy> getListByStockList(int num,String[] stockIdList); | | | | | |
| 3 | 简要说明 | **//模式二: 持仓匹配程度搜索，基于客户 持仓股占比期望集合 进行匹配** | | | | | |
| 函数申明 | List<FundHeavy> getListByStockScore(int num,String[] stockIdList,String[] stockRadioList); | | | | | |
| 4 | 简要说明 | **//模式三: 板块搜索简版，不考虑股票占比，按板块匹配数与集中度进行匹配** | | | | | |
| 函数申明 | List<FundHeavy> getListByStockAllType(int num,String[] TypeList); | | | | | |
| 5 | 简要说明 | **//模式三加: 板块搜索，考虑股票占比以及按板块匹配数与集中度进行匹配** | | | | | |
| 函数申明 | List<FundHeavy> getListByStockAllTypeRadio(int num,String[] TypeList); | | | | | |
| **模式三加的的相关步骤** | **1//注意这里的fundHeavy是单例，所以要清零** | | |  | | |  |
| **2//对List评分** | | |  |
| **3//用户选择多种板块纳入考量** | | |
| **4//统计各持仓板块集中度** | | |
| **5 //排序** | | |
| **6 //这里有可能不满十个基金项，甚至一个都没有的情况，减小规模** | | |
| **7 //打包数据给controller层** | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 后端请求与股票信息更新功能  @Service public class HttpClient  @RequestMapping("/oneStock") public String oneStock (String id) | | | | | |
|  |  | 返回类型 | 方法名 | 参数 | 说明 |
| 1 | 简要说明 | **//后端请求功能** | | | |
| 函数申明 | public String client(String url, HttpMethod method,MultiValueMap<String,String>params) | | | |
| 2 | 简要说明 | **//更新股票信息** | | | |
| 函数申明 | public String oneStock (String id) | | | |

|  |  |
| --- | --- |
| 以及用户对象方法描述与股票对象方法描述等等 | |
|  |  |

## 3.4 系统用例实现详细设计

### 3.4.1总用例图

### 3.4.2顺序图

|  |
| --- |
| 1股票普通搜索 |
|  |
| 2基金普通搜索 |
|  |
| 3股票详细信息 |
|  |
| 4基金详细信息（重仓股信息） |
|  |
| 5持仓搜索（不参考相关重仓股比例） |
|  |
| 6持仓搜索（需参考相关重仓股比例） |
|  |
| 7板块搜索（不限定某一公司产品） |
|  |
| 8心仪基金收藏 |
|  |
| 9股票详细信息页查询 |
|  |
| 10基金详细信息页查询 |
|  |

## 系统非功能设计

### 3.5.0 约束矩阵

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 模块  约束 | 账户模块 | 搜索模块 | 用户管理基金模块 | 股票和基金管理模块 | 系统管理模块 | 统计模块 |
| 性能设计 | √ | √ | √ | √ |  |  |
| 可维护性设计 | √ | √ | √ | √ | √ | √ |
| 易用性设计 | √ | √ | √ | √ | √ |  |
| 兼容性设计 | √ | √ | √ |  |  |  |
| 可拓展性设计 | √ | √ | √ | √ | √ | √ |

### 3.5.1 性能设计

* + - 系统响应速度：在非高峰时间根据编号和名称特定条件进行搜索，可以在3秒内得到搜索结果。
    - 当通过互联网接入系统的时候，期望在编号和名称搜索时最长查询时间<15秒。
    - 满足至少100个以上的并发用户数
    - 数据存储量：每年的数据存储容量不超过1（G），未来五年该容量的预期增长值为1（G）。未来累计存储容量不超过9（G）.

### 3.5.2 维护性设计

* 数据库的设计要满足业务数据的扩充以及满足新数据增加和扩充数据体系的要求
* APP具有较高的自动化程度，如：自动异常调度、自动故障告警、自动任务恢复等
* 系统具有良好的扩充、发展能力，提供今后扩充系统功能、规模的接口
* 系统发生故障后能够排除(或抑制)故障予以修复，并返回到原来正常运行状态
* 定期备份指定范围的数据，或根据需要随时备份，以便在需要的时候将备份的数据恢复。并且能够通过设定，利用系统提供的自动通知功能，提醒系统管理人员备份数据。具备相应容错手段，系统管理员能根据某一时间点进行数据的恢复，维护管理数据库死锁问题和维护数据库内数据的一致性

### 3.5.3 易用性设计

* 该APP提供基于图形化的友好管理界面
* 用户的任何操作需要即时给出反馈
* 提供撤销和重做的功能，注意区分功能的优先级
* 帮助用户识别、诊断、并从错误中恢复，将损失降到最低。如果无法自动挽回，则提供详尽的说明文字和指导方向
* 使用用户易懂和约定俗成的表达方式，而不是专业术语
* 符合IE界面风格：

 系统风格协调一致，体现人性化的友好性管理界面

 符合用户习惯或者容易被用户接受的管理风格

 具有容错能力，包括错误诊断和提示，并具有快速的系统反应的管理界面

* 提供详细的、易懂的联机帮助信息，所有的操作菜单和提示信息全部使用中文，协助用户使用

### 3.5.4 兼容性设计

* 支持不同Android版本

### 3.5.5 可拓展性设计

* 支持后期用户数量增大时引起的应用需求
* 可实现负载均衡；日后若信息量较大，则系统可相应增加服务器实现扩展。
* 支持基金搜索和股票搜索的业务可拓展性，提供扩展定制开发能力
* 支持纵向拓展：也就是支持通过软件升级或者更换更高性能硬件来提升系统性能
* 支持横向拓展：也就是支持使用更多的服务器来提高性能

### 3.5.6 可靠性设计

* 系统满足7\*24稳定运行需要。系统平均无故障时间（MTTF）大于4320小时（180天）
* 系统容错性强，在外系统故障、用户非法操作、数据内容/格式出错的情况下，仍可正常运行平均失效间隔（MTBF）大于4320小时（180天）
* 系统故障状态下恢复时间短，平均恢复时间（MTTR）小于10分钟
* 对人工输入的数据以及来自不同接口的数据进行合法性检查，对错误数据进行自动纠错处理

### 3.5.7 灾备设计

* 系统做到能支持定期的自动数据备份操作，还提供人工备份的操作
* 提供多种数据输出格式，该输出格式可以方便快捷的倒回原来系统中
* 真正做到使系统能在数据损坏，丢失等情况下将备份数据倒回，实现数据恢复

### 3.5.8 安全性设计

* 用户管理，每个用户只能访问本用户相关的信息，不得逾越
* 权限管理，用户或者计算机程序只能访问当下所必需的信息或者资源。
* 对数据范围权限进行控制
* 防止对数据的篡改和攻击
* 对每次访问都必须经过安全检查审核
* 提高整个系统安全最低点的安全性能
* 在系统被攻击后应该尽快回复信息中心的功能，减少损失
* 安全系统的设计应该具有一致性，以达到安全信息共享的要求