**理财产品管理系统**

**概要设计说明书**

不想编码就去理财队

修订历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修订日期 | 修订内容 | 修订人 | 版本号 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 概述

## 系统简述

## 软件设计目标

### 功能性设计目标

对需求进行优先级的划分

### 非功能性设计目标

对非功能需求的总结

## 参考资料

# 术语表

统计一些在项目过程中可能会发生歧义、一定要清晰定义的术语

# 设计概述

## 系统结构设计

### 系统逻辑架构

### 系统物理架构

### 系统逻辑数据模型

## 系统功能划分与设计

### 子系统-模块名字

#### 功能描述

#### 各个用例的流程图

#### 概念类图

## 系统接口设计

### 提供给用户的接口

各个网页

### 系统内部的接口

各个模块在工作的时候需要用到的接口,不需要具体到参数类型等详细内容,但是起码给接口一个名字

|  |  |  |
| --- | --- | --- |
| 接口描述 | 所属模块 | 备注 |
| 从产品标准库中获取产品信息 | 产品配置 |  |
| 从配置表中获取配置信息 | 产品配置 |  |
| 将已配置产品添加到待产品配置入库审批表 | 产品配置 |  |
| 从待产品配置入库审批表获取产品信息 | 产品配置 |  |
| 将已审批通过产品添加到配置库 | 产品配置 |  |

### 系统外部的接口

无

## 约束和假定

约束:

时间日程

项目质量

资金预算

假设:

比如说我们都是全栈工程师

项目三个约束条件(Triple Constraint)：时间日程（项目周期）、性能规规（质量）、以及资金（成本费用）预算。项目的假设没找到具体定义。

约束条件和假设是可以贯穿整个项目的。比如，项目必须在10月份完成（约束条件），预算是100万（约束条件）。我们提供5个熟练的队员（假设有5个人，假设他们是熟练的）。到后期，5个人来了4个（这时就不是假设了，是问题了）。4个人中，还有一个菜鸟（又成问题了）。假设也是风险的来源，如果不成立，就会转为问题。

## 非功能性设计