<丰收宝盒>

补充规约

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 11/9/2024 | 1.0 | 第一次修正，初始设计 | 侯杰 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 定义、首字母缩写词和缩略语 4

1.4 参考资料 4

1.5 概述 4

2. 功能 4

2.1 <用户端功能> 4

2.2 <管理端功能> 5

3. 可用性 6

4. 可靠性 7

5. 性能 7

6. 可支持性 8

7. 设计约束 9

8. 联机用户文档和帮助系统需求 11

9. 购买的构件 11

10. 接口 11

10.1 用户界面 11

10.2 硬件接口 11

10.3 软件接口 11

10.4 通信接口 12

11. 许可需求 12

12. 法律、版权及其他声明 12

13. 适用的标准 12

补充规约

# 简介

## 目的

#### 本补充规约文档的目的在于详细描述系统中无法通过用例模型直接捕捉的需求。这些需求对于系统的整体功能、性能和合规性至关重要，确保系统符合相关法律、法规和标准，并且在可用性、可靠性、性能和可支持性等质量属性方面满足预期。

## 范围

此补充规约涵盖了“丰收宝盒”高品质农产品甄选平台系统的非功能性需求和法律、技术约束。其范围包括系统的质量属性、兼容性要求、设计约束，以及与外部环境的集成要求。文档适用于管理端和用户端的整体系统开发和维护。

## 定义、首字母缩写词和缩略语

* **SKU**：库存单位 (Stock Keeping Unit)，指可库存管理的最小商品单元。
* **SPU**：标准产品单元 (Standard Product Unit)，商品类别的标准化信息。
* **UI**：用户界面 (User Interface)，指用户与系统交互的界面。
* **API**：应用程序接口 (Application Programming Interface)，用于系统与外部模块或系统之间的通信。

## 参考资料

* 法律法规文件：针对电子商务和数据隐私的相关法律法规（如《电子商务法》、《数据安全法》）。

## 概述

该补充规约的后续部分将详细列出系统的以下需求：

* **法律和法规上的需求**：系统需遵循的法律标准和行业规范。
* **质量属性**：包括可用性、可靠性、性能和可支持性需求，确保系统具有良好的用户体验和运维能力。
* **其他需求**：涵盖操作系统及环境的要求，兼容性需求，设计约束，确保系统在不同环境下的正常运行。

# 功能

本节旨在以自然语言描述“丰收宝盒”高品质农产品甄选平台系统的核心功能需求。为了便于组织和理解，本部分将按系统的主要特性和用户类型进行划分，涵盖了用户和管理端的各个功能模块。

## <用户端功能>

##### 2.1.1 商品浏览与搜索

* **需求说明**：用户可以通过平台浏览并搜索各类农产品，商品按照类别、价格、销量等进行排序和过滤。
* **功能要点**：
  + 支持按商品分类和属性（如产地、品牌、价格范围等）进行筛选。
  + 提供详细的商品描述页面，展示商品图片、规格、评价等信息。

##### 2.1.2 购物车管理

* **需求说明**：用户可以将商品添加至购物车，并对购物车中的商品进行管理。
* **功能要点**：
  + 支持添加、删除商品，以及修改商品数量。
  + 实时显示商品价格总计，支持优惠券的使用。
  + 检查库存情况并提醒用户商品的库存状态。

##### 2.1.3 订单管理

* **需求说明**：用户可以创建、查看和管理订单，跟踪订单的处理进度并进行支付。
* **功能要点**：
  + 支持多种支付方式（如微信支付、银行卡支付等）。
  + 提供订单状态跟踪功能。
  + 允许用户取消订单、申请退款或售后服务。

##### 2.1.4 地址管理

* **需求说明**：用户可以添加、编辑和删除收货地址。
* **功能要点**：
  + 支持多地址管理，用户可以在结算时选择不同的地址。
  + 自动检测地址的格式错误，并提供纠正提示。

##### 2.1.5 个人信息管理

* **需求说明**：用户可以维护个人账户信息，包括联系方式、用户名等。
* **功能要点**：
  + 支持修改用户名、密码，确保账户安全。

## <管理端功能>

##### 2.2.1 商品管理（SPU/SKU管理）

* **需求说明**：商家通过管理端对商品信息进行管理，包括标准产品单元（SPU）和库存管理单元（SKU）。
* **功能要点**：
  + 支持新增、编辑、删除商品信息。
  + SKU管理允许商家管理库存、规格、价格等信息。

##### 2.2.2 订单管理

* **需求说明**：商家可以处理并跟踪订单，从订单生成到发货全过程的管理。
* **功能要点**：
  + 提供订单状态更新功能，商家可以标记订单为已发货、已完成等状态。
  + 支持订单查询和筛选，按日期、状态、客户等条件筛选订单。

##### 2.2.3 员工管理

* **需求说明**：管理平台的工作人员可以管理员工信息和操作权限。
* **功能要点**：
  + 支持创建、编辑员工账户，分配不同的角色和权限。
  + 提供员工操作日志，记录员工的系统操作行为。

##### 2.2.4 分类管理

* **需求说明**：商家可以创建和管理商品分类，确保商品展示的有序性。
* **功能要点**：
  + 支持多级分类管理。
  + 提供分类编辑和删除功能。

##### 2.2.5 数据统计与分析

* **需求说明**：平台提供数据统计和可视化工具，帮助商家进行决策分析。
* **功能要点**：
  + 提供销售数据、订单数据的统计报表。
  + 支持数据导出功能，供商家进行进一步分析。

# 可用性

可用性需求旨在确保系统能够提供用户友好且高效的操作体验。以下需求概述了用户在使用系统时的培训、任务执行效率和界面标准等方面的要求。

#### 3.1 培训时间

* **需求说明**：普通用户和高级用户在使用系统时，所需的学习和培训时间应最小化。
  + **普通用户**：通过微信小程序使用该平台的用户应能够在 **10-15 分钟** 内熟悉商品浏览、购物车管理、订单操作等基本功能，无需专业培训。
  + **高级用户（商家）**：商家需要对系统管理端的使用进行培训，商家应在 **2-3 小时** 内掌握商品管理、订单管理和数据分析功能。

#### 3.2 任务执行效率

* **需求说明**：系统中的常见任务应在合理的时间内完成，以提升用户体验。
  + 普通用户进行商品浏览、添加商品到购物车、下单等任务应在 **1-3 分钟** 内完成。
  + 管理端用户应能在 **5 分钟** 内完成一次库存商品的更新或订单的处理操作。

#### 3.3 界面标准

* **需求说明**：系统界面设计应符合公认的可用性标准，以确保用户体验一致。
  + 界面设计应符合 **Microsoft GUI 标准** 和 **IBM CUA 标准**，确保系统具备易用性、一致性和直观的操作流程。
  + 界面布局应适应移动端（微信小程序）的操作习惯，支持多语言版本（如中文、英文），以提高用户的国际化体验。

# 可靠性

系统的可靠性需求确保系统能够在规定的时间范围内高效、持续地运行，并具备故障处理和恢复能力。

#### 4.1 平均故障间隔时间 (MTBF)

* **需求说明**：系统的平均故障间隔时间应足够长，以确保系统高效稳定地运行。
  + 系统的 **MTBF** 应不低于 **2000 小时**（约为 83 天），即系统平均每隔 **2000 小时** 才会出现一次故障。

#### 4.2 平均修复时间 (MTTR)

* **需求说明**：当系统发生故障时，修复时间应尽量缩短，减少系统停机对用户的影响。
  + 系统的 **MTTR** 应不超过 **2 小时**，即从故障发生到完全恢复的时间不应超过 **2 小时**。

#### 4.3 精确度

* **需求说明**：系统输出的数据应具备高精确度，特别是在订单、库存和统计数据方面。
  + 订单和库存管理系统的数据精确度应达到 **99.99%**，即每处理一万笔订单或库存交易，出错不超过 **1 笔**。
  + 数据统计报表中的数值偏差应不超过 **0.1%**，以确保商家决策的可靠性。

#### 4.4 最高错误或缺陷率

* **需求说明**：系统代码的错误率应维持在行业标准范围内，避免重大缺陷影响用户体验。
  + 系统的最高错误率应低于 **0.5 bugs/KLOC**（每千行代码的错误数目），即每千行代码中最多允许 **0.5 个错误**。
  + 功能模块的错误率应控制在 **0.2 bugs/function-point**，确保系统的每个功能点在设计和实现时的错误率不超过 **0.2**。

# 性能

本节描述“丰收宝盒”高品质农产品甄选平台的性能需求，确保系统在高效运行的同时，满足用户的响应速度、吞吐量、容量和资源利用等方面的要求。

#### 5.1 响应时间

* **需求说明**：系统在处理用户请求时，应具备快速响应能力，以确保用户体验的流畅性。
  + **平均响应时间**：对于普通用户的操作（如浏览商品、添加到购物车、查看订单），响应时间应不超过 **2 秒**。
  + **最长响应时间**：在系统高负荷情况下，响应时间不应超过 **5 秒**，例如在高峰购物时段如促销或节假日。
  + **支付操作**：由于涉及外部支付接口，支付确认操作的响应时间应控制在 **5 秒** 内。

#### 5.2 吞吐量

* **需求说明**：系统需要在高并发情况下保证处理能力，以满足大规模用户同时使用的需求。
  + **每秒事务处理能力（TPS）**：系统应能够支持 **1000 次/秒** 的事务处理能力，包括商品查询、订单创建等操作。
  + **峰值流量**：在高峰期（如双十一、丰收季等活动期间），系统应支持每秒 **3000 次/秒** 的事务处理。

#### 5.3 容量

* **需求说明**：系统需要能够容纳一定数量的并发用户和处理事务。
  + **用户容量**：系统应支持 **50 万** 注册用户并发使用，并允许 **10 万** 用户同时在线。
  + **订单处理容量**：系统应每日至少处理 **10 万笔** 订单，并且在高峰期（如促销日）能够处理 **30 万笔/天** 订单。

#### 5.4 降级模式

* **需求说明**：当系统部分功能出现问题时，应具备降级运行的能力，确保用户的关键操作不受影响。
  + **可接受的运行模式**：
    - 若订单处理系统故障，用户仍可浏览商品、添加至购物车，但无法进行结算和支付，系统将提示延迟处理。
    - 若支付系统故障，订单仍可创建，并支持用户在恢复后支付，订单状态显示为“待支付”。
    - 数据统计和后台管理系统可以暂停运行，但用户前台功能应不受影响。

#### 5.5 资源利用情况

* **内存**：系统的后台服务应在高并发情况下，每个实例的内存使用量控制在 **2GB** 以内，防止内存溢出。
* **磁盘**：系统需要高效的磁盘读写性能，尤其是在处理大量订单数据时。每日磁盘使用增长应不超过 **1GB**，并采用自动备份和归档机制。
* **网络通信**：在网络资源的使用上，系统应通过优化通信协议，确保每次请求的数据包大小控制在 **10KB** 以下，最大限度减少带宽占用。

# 可支持性

本节描述了为了提高“丰收宝盒”高品质农产品甄选平台的可支持性和可维护性而需遵循的要求。这些要求包括编码标准、命名约定、使用的类库以及维护相关的访问和工具，确保系统在开发、维护和更新过程中易于管理、扩展和调试。

#### 6.1 编码标准

* **需求说明**：系统的开发应遵循统一的编码标准，以提高代码的可读性和可维护性。
  + 采用 **Google Java Style Guide** 作为 Java 代码的规范标准。
  + 所有代码需经过代码审查和自动化工具检测，确保符合标准，避免出现难以维护的冗余代码或复杂逻辑。
  + 变量、函数和类的命名需使用 **CamelCase** 命名法，且变量名应具备描述性，避免使用无意义的缩写。

#### 6.2 命名约定

* **需求说明**：系统中所有资源（包括数据库表、字段、接口、文件名等）应遵循一致的命名约定，避免因命名混乱导致的维护困难。
  + **数据库表命名**：所有表名应使用小写字母，并以“tbl\_”前缀开头，使用下划线分隔单词，例如：tbl\_order\_details。
  + **API 命名**：API 路由应简洁明了，使用 RESTful 风格，例如：/api/v1/orders、/api/v1/products/{id}。
  + **文件命名**：所有项目文件（如配置文件、脚本）应以描述性名称命名，避免使用临时名称或缩写，如 config\_payment\_gateway.yaml。

#### 6.3 类库和框架使用

* **需求说明**：系统开发中使用的第三方类库和框架应经过充分验证，避免使用过时或低质量的类库，以提高系统的可维护性和安全性。
  + 优先使用主流且活跃维护的类库和框架，例如 **Spring Boot**（用于后台开发）、**MyBatis**（用于数据库操作）和 **Bootstrap**（用于前端开发）。
  + 第三方类库应定期更新，确保系统安全性和功能的持续改进。
  + 使用 **Maven** 来管理项目依赖，并避免类库版本冲突。

#### 6.4 维护访问权

* **需求说明**：为了确保系统的可维护性，维护人员需要具备对系统的充分访问权限和操作工具。
  + 后台管理系统应提供维护人员的专用访问端口，允许进行系统日志查看、性能监控和数据备份操作。
  + 通过 **SSH** 安全访问服务器，维护人员应具备对生产环境和测试环境的访问权限。
  + 提供自动化部署工具（如 **Jenkins**）和容器化平台（如 **Docker**），确保系统的快速部署和版本更新。

#### 6.5 维护工具和实用程序

* **需求说明**：系统需配备相关的维护工具和实用程序，便于维护人员排查问题和优化系统性能。
  + **日志系统**：集成 **ELK（Elasticsearch, Logstash, Kibana）** 作为日志管理和监控平台，便于快速定位和解决系统故障。
  + **数据库管理工具**：提供数据库备份和恢复工具，定期对数据库进行维护和优化，避免数据积压或索引失效导致的性能下降。
  + **性能监控工具**：使用 **Prometheus** 和 **Grafana** 实时监控系统性能指标（如 CPU 使用率、内存消耗、网络延迟等），以便在问题出现前进行预防性维护。

# 设计约束

本节列出了“丰收宝盒”高品质农产品甄选平台在开发过程中需遵循的设计约束。这些约束是已被批准并在系统设计和实现中必须遵守的决策，确保系统在语言、框架、工具和架构等方面的标准化和一致性。

#### 7.1 开发语言

* **需求说明**：平台的开发必须使用指定的编程语言，以确保团队成员的熟悉度和代码一致性。
  + **后端**：后端服务必须使用 **Java** 编程语言，基于 **Spring Boot** 框架进行开发。
  + **前端**：前端用户界面应使用 **HTML5**、**CSS3** 和 **JavaScript**，并使用 **Vue.js** 框架来构建单页面应用（SPA）。
  + **小程序**：由于系统将在微信小程序中运行，前端小程序部分必须使用 **微信小程序框架**，并采用 **WXML** 和 **WXSS** 进行页面布局和样式设计。

#### 7.2 数据库选择

* **需求说明**：系统的数据存储必须采用已批准的数据库技术，确保数据管理的一致性和性能。
  + **关系型数据库**：系统应使用 **MySQL** 作为主要的关系型数据库，用于存储商品、订单、用户等核心业务数据。
  + **NoSQL 数据库**：为了优化缓存性能，系统需使用 **Redis** 作为缓存层，减少数据库压力，提升用户查询响应速度。
  + **数据库结构**：数据库设计需遵循 **第三范式** 进行规范化，确保数据的一致性和可维护性。

#### 7.3 软件流程需求

* **需求说明**：系统的开发和发布流程必须严格遵循既定的软件开发流程，以确保开发质量和一致性。
  + **开发流程**：系统采用 **敏捷开发（Agile）** 方法论，实行两周为周期的迭代开发，每个迭代结束时进行功能发布和评审。
  + **代码管理**：使用 **Git** 进行版本控制，所有代码必须经过代码审查（Code Review）并通过自动化测试后才能合并至主分支。
  + **持续集成与部署**：系统必须使用 **Jenkins** 进行持续集成（CI），并通过 **Docker** 容器化部署，确保快速发布和回滚。

#### 7.4 架构约束

* **需求说明**：系统的整体架构设计需遵循既定的分层和模块化原则，以确保系统的可扩展性和维护性。
  + **微服务架构**：系统采用 **微服务架构（Microservices Architecture）**，将不同业务模块（如订单管理、商品管理、用户管理）拆分为独立的微服务，以提高系统的可扩展性和可靠性。
  + **API 网关**：所有前端请求必须通过 **API 网关** 进行路由和负载均衡，确保后端微服务的统一访问入口和安全性。
  + **消息队列**：使用 **RabbitMQ** 或 **Kafka** 作为异步消息队列，处理订单异步通知和用户行为数据分析等任务，确保系统在高并发场景下的稳定性。

#### 7.5 开发工具

* **需求说明**：系统开发过程中必须使用指定的开发工具，以统一团队的开发环境并提升协作效率。
  + **IDE**：推荐使用 **IntelliJ IDEA** 作为主要开发工具，支持 Java 和 Vue.js 开发，统一团队开发环境。
  + **构建工具**：使用 **Maven** 管理后端项目的构建和依赖，前端项目则使用 **npm** 进行包管理和构建。
  + **调试工具**：系统的开发和调试阶段需使用 **Postman** 或 **Swagger** 进行 API 接口调试，确保接口文档和实际实现的一致性。

# 联机用户文档和帮助系统需求

本节描述了系统联机文档的要求，确保用户和管理员能够快速理解和使用系统功能。

#### 8.1 联机用户文档

* **需求说明**：系统提供全面的在线用户文档，帮助用户了解如何使用平台的各项功能。文档需包括使用指南、常见问题解答和详细操作步骤，覆盖用户端和管理端。
  + 用户文档应包括商品浏览、购物车管理、下单支付、订单跟踪、个人信息管理等操作的详细说明。
  + 管理员文档应包括商品管理、库存管理、订单处理、数据分析等后台功能的操作指南。
  + 所有文档需支持中、英文版本，并提供基于关键词的搜索功能。

# 购买的构件

本节列出了系统中使用的所有购入构件以及相应的许可和限制，确保这些构件与系统兼容，并符合使用规定。

#### 9.1 数据分析工具

* **需求说明**：系统可使用第三方数据分析工具，如 **Google Analytics** 或 **百度统计**，用于收集和分析用户行为数据。
  + 需遵守这些服务的隐私和数据使用协议，确保用户数据的合法使用。

# 接口

本节规定了“丰收宝盒”系统所需支持的各类接口，包括用户界面、硬件接口、软件接口和通信接口。

## 用户界面

* **用户端界面**：设计需简洁明了，采用响应式设计，确保在移动端和桌面端都具有良好的用户体验。小程序界面符合微信小程序的设计规范。
* **管理端界面**：管理员界面支持多层次数据管理和可视化报表展示，功能模块划分清晰，提供易于操作的商品、订单和用户管理功能。
* 界面支持特色皮肤切换，满足不同使用者的个性化需求。

## 硬件接口

无

## 软件接口

* **支付接口**：与微信支付和支付宝支付的接口需遵循其官方 SDK 规范，通过 HTTPS 安全协议进行数据传输。
* **物流接口**：顺丰和京东物流的 API 应使用 RESTful 风格，通过 HTTP POST 和 GET 请求实现订单的物流查询和追踪。
* **数据分析接口**：与 Google Analytics 或百度统计的数据接口需通过 JavaScript 嵌入代码，实时跟踪用户行为数据。

## 通信接口

* **需求说明**：系统支持与外部系统和设备进行数据通信。
  + **网络接口**：系统应通过标准的 HTTP/HTTPS 协议与客户端和外部服务通信，确保数据加密传输。
  + **消息队列接口**：系统内的异步任务应使用 **RabbitMQ** 或 **Kafka** 进行消息传递，保证高并发时的可靠性。
  + **API 接口**：系统需提供 RESTful API，支持与外部系统进行集成，如 ERP 系统、支付系统和数据分析工具。

# 许可需求

* 开源库需符合其许可证（如 **MIT**、**GPL**）的要求，确保合法使用和发布。
* 支付、物流等第三方服务的使用需遵循其服务协议，确保不会违反其许可规定。

# 法律、版权及其他声明

* + 平台在页面显著位置注明 **版权声明** 和 **隐私政策**，确保用户知悉其数据的使用方式。
  + 系统涉及的所有支付、物流服务应遵循相关的 **电子商务法** 和 **隐私保护法**，确保合法运营。

# 适用的标准

* + **软件开发标准**：系统开发应遵循 **ISO/IEC 12207** 软件生命周期标准，确保开发过程的规范化。
  + **安全标准**：系统需符合 **ISO/IEC 27001** 信息安全管理标准，确保用户数据和交易数据的安全性。
  + **隐私标准**：系统需遵循 **GDPR（欧盟通用数据保护条例）** 和 **中国网络安全法** 的相关规定，确保用户隐私数据的保护。
  + **国际化标准**：系统需遵循 **Unicode** 字符编码标准，以确保多语言支持的准确性和兼容性。