

需求文档：订单管理系统

1. 概述

该订单管理系统旨在有效地处理订单流程，包括创建订单、处理出库和入库任务，以及异常订单的处理。以下是系统主要功能和相关业务流程的详细描述。

2. 订单管理

2.1 订单表 (Orders)

1. 创建订单

- 接收订单数据，包括用户信息和商品明细。
- 验证用户信息和库存是否足够。
- 扣减库存并生成订单。
- 记录订单相关操作，初始化订单状态为"Pending"。

2. 更新订单状态

- 处理订单，根据业务需要更新订单状态。
- 发货，当订单状态为"ReadyForShipment"时，进一步更新为"Shipped"。
- 异常处理，如果出现问题，更新状态为"Exception"，并记录异常详细信息。

3. 取消订单

- 检查可取消性，某些状态下的订单可能不允许取消。
- 如果可以取消，更新订单状态，并将库存数量回滚。

4. 订单查询

- 按用户查询，允许用户查询自己的订单历史。
- 按状态查询，允许管理员根据订单状态进行查询。

2.2 订单商品表 (OrderItems)

1. 添加商品到订单

- 验证商品存在性和库存情况。
- 创建订单商品条目并关联到具体订单。

2. 更新订单商品

- 更改数量，同时更新库存。
- 删除商品，从订单中移除商品，并更新库存。

3. 查询订单商品

- 按订单查询，查询某个订单中的所有商品条目。
- 商品详情，包括数量和商品信息。

2.3 订单异常表 (OrderExceptions)

1. 记录异常

- 创建异常记录，包括详细描述、订单ID和创建时间。
- 关联订单，确保每个异常记录都能指向一个具体订单。

2. 更新异常记录

- 更新描述，根据问题解决进程更新异常描述或状态。
- 关闭异常，一旦问题解决，更新异常记录为已解决状态。

3. 查询异常

- 按订单查询异常，查询特定订单的所有相关异常。
- 按日期查询，按创建日期查询异常记录。

3. 任务管理

3.1 入库任务表 (InboundTasks)

1. 创建入库任务

- 初始化任务，设置状态为"Pending"。
- 记录创建时间。

2. 更新任务状态

- 分配任务，当任务被分配给特定的仓库员工时，更新状态为"Assigned"。
- 完成任务，更新状态为"Completed"。

3. 删除或取消任务

- 取消未开始的任務，如果入库任务还未开始，可以将其取消。

4. 查询入库任务

- 按状态查询，查询不同状态的入库任务。
- 按日期范围查询，查询特定日期范围内创建或更新的任务。

3.2 出库任务表 (OutboundTasks)

1. 创建出库任务

- 初始化任务，当订单状态变为"ReadyForShipment"时创建，设置状态为"Pending"。
- 记录任务和订单关联。

2. 更新任务状态

- 分配任务，当任务被分配给特定的物流操作员时，更新状态为"InProgress"。
- 完成任务，更新状态为"Completed"。
- 处理异常，如遇到任何问题，记录异常并更新状态。

3. 查询出库任务

- 按状态查询，查询不同状态的出库任务。
- 按日期查询，查询特定日期范围内创建或更新的出库任务。

4. 货物管理

4.1 入库货物表 (InboundItems)

1. 添加入库货物

- 验证产品存在性。
- 创建入库货物记录，包括数量和生产日期。
- 关联入库任务。

2. 更新入库货物

- 调整数量和更新生产日期。

3. 删除入库货物

- 删除记录，如果入库货物被错误记录或入库任务被取消。

4. 查询入库货物

- 按任务查询，查询特定入库任务关联的所有货物。
- 按产品查询，查看特定产品的入库记录。

4.2 出库货物表 (OutboundItems)

1. 添加出库货物

- 验证产品和任务存在性。
- 创建出库货物记录，包括数量。
- 关联出库任务。

2. 更新出库货物

- 调整数量。

3. 删除出库货物

- 删除记录，如果出库货物被错误记录或出库任务被取消。

4. 查询出库货物

- 按任务查询，查询特定出库任务关联的所有货物。
- 按产品查询，查看特定产品的出库记录。

5. 产品管理

5.1 产品表 (Products)

1. 添加产品信息

- 添加数据库中的产品信息。

2. 获取产品信息

- 从缓存中检索产品信息；如果缓存未命中，则从数据库获取并更新缓存。

3. 更新产品信息

- 更新数据库中的产品信息，并更新缓存中相应的数据。

4. 删除产品信息

- 删除数据库中的产品信息。

5. 删除产品缓存

- 在产品信息被更新或需要强制刷新时删除缓存。

6. 库存管理

6.1 库存表 (Inventory)

1. 更新库存

- 增加库存，当接收到入库货物时。
- 减少库存，当订单中的商品被确认时。

2. 查询库存

- 按产品查询，查询特定产品的

当前库存量。

- 库存警告，标识库存量低于安全水平的产品。

3. 定时更新库存状态

- 定时任务，定期检查库存数据，包括检查过期产品，更新最后库存时间等。

7. 技术要求

- 数据库：建议采用关系型数据库，如MySQL。
- 缓存：使用缓存技术提高系统性能，可考虑使用Redis等。
- 后端开发：选择合适的后端开发框架，如Spring Boot（Java）等。

8. 安全要求

- 访问控制：确保用户只能访问其权限范围内的数据和功能。
- 数据加密：对敏感数据进行加密传输，如用户密码、支付信息等。
- 安全审计：记录用户操作日志，以便审计和追踪问题。

9. 性能要求

- 响应时间：系统应保证快速响应用户请求，特别是订单处理等核心功能。
- 并发处理：能够处理大量并发请求，保证系统的稳定性和可用性。