添加工厂（名称 地址等基本信息） 设置为主工厂 添加商品（当前工厂的商品编码 商品名称） 配置编码关系 添加仓库

添加多个工厂之后，配置工厂商品之间编码关系

选择乙工厂—商品编码 == 主工厂—商品编码（前提是主工厂的编码大于等于分工厂之和）

添加客户信息

制布工艺？密度？幅宽？颜色？码口？码数？

添加入库清单 工厂 时间 商品编码 商品详细信息 单价 仓库 目标仓库自动更新该商品数量

添加出库清单 客户 工厂 时间 商品编码 商品详细信息 单价 仓库 目标仓库自动更新该商品数量

查看库存 仓库调配 原仓库中的商品调配到目标仓库中

查看库存 查看某一类商品的流水记录和调配记录 查看对应出入库单

工厂表：

自增ID 工厂名称 工厂地址 工厂基本信息 主工厂标识（自己的工厂1 主制布工厂2 分工厂3）

商品表：

（主工厂的话同时插入两个编码，非主工厂只插入商品编码，配置编码时更新主编码）

自增ID 工厂ID 商品主编码 商品编码 商品名称

/\*\*\*商品详情表：

自增ID 商品详细信息\*\*\*/

仓库表：

自增ID 工厂ID 仓库名称 仓库基本信息 仓库最大容量

库存表：

自增ID 仓库ID 工厂ID 商品编码 商品详细信息 数量

流水记录表：（记录出入库）

自增ID 工厂ID 仓库ID 商品编码 商品详细信息 出入库标志 出入库对应时间戳

调配记录表：（记录调配历史）

自增ID 原工厂ID 原仓库ID 原商品编码 目标工厂ID 目标仓库ID 目标商品编码 商品详细信息 调配数量 调配时间

客户表：

自增ID 客户名称 客户基本信息

入库表：

自增ID 时间戳 工厂ID 仓库ID 进货时间 商品编码 商品详细信息 单价 合计

出库表：

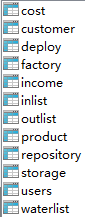
自增ID 时间戳 客户ID 工厂ID 仓库ID 出货时间 商品编码 商品详细信息 单价 合计

成本表：

自增ID 入库时间戳 合计 完成支付标识

收入表：

自增ID 出库时间戳 合计 完成支付标识



假设 总价 = 单价 \* 码数

入库的插入→流水表&库存表的插入&成本表的插入

出库的插入→流水表的插入&库存表的修改&收入表的插入

入库的删除→流水表的删除&库存表的修改&成本表的删除

出库的删除→流水表的删除&库存表的修改&收入表的删除

入库的修改

1.修改了码数→库存表的修改&成本表的修改

2.修改了商品详情→流水表的修改

3.修改了单价→成本表的修改

4.修改了工厂/仓库→库存表的修改

出库的修改

1.修改了码数→库存表的修改&成本表的修改

2.修改了商品详情→流水表的修改

3.修改了单价→成本表的修改

4.修改了工厂/仓库→库存表的修改

入库的退货→库存表的修改&成本表的修改

出库的退货→库存表的修改&收入表的修改

库存 成本 收入 流水 页面只能查，增加一个显示详细的页面

客户名称 输入框切换成select2里面的输入筛选框 √

库存 流水 都增加一个商品主编码 √

出库的时候增加库存查询

成本 收入 流水单 详情页面是同一个页面结构

成本点详情 成本 入库 页面传递两个实体类

收入点详情 收入 出库 页面传递两个实体类

流水点详情 传参 timestamp 出入库标志 判断走出库还是入库的页面 然后同成本或收入

编码管理

添加：主工厂 => 主编码 匹配 分工厂 => 编码

调配管理

查询：写两个sql，一个调出一个调入，查全部的时候先查调出，再查调入，按照时间来排序（单号顺序越大时间越近）

添加：

查看详细之后才能删除，或者修改或者打印

添加调配 = 出入库操作的增加

增加一个库存查询

StorageService 先从调出的仓库里减去一定数量，然后再查调入的仓库之前是否有商品，有的话就增加没有就创建

删除调配 = 出入库操作的删除

从调出的仓库里增加数量，从调入的仓库里减少数量，还得判断是否有商品

修改调配 = 出入库操作的修改