

第一次项目答辩

超市管理系统



2023

演讲人：岳新亮

团队：MYHD

CONTENTS

目录



01 | 项目概览
Review of work content

02 | 代码编写思路
Market data analysis

03 | 代码难点
Work experience summary

04 | 项目展示
Future work planning

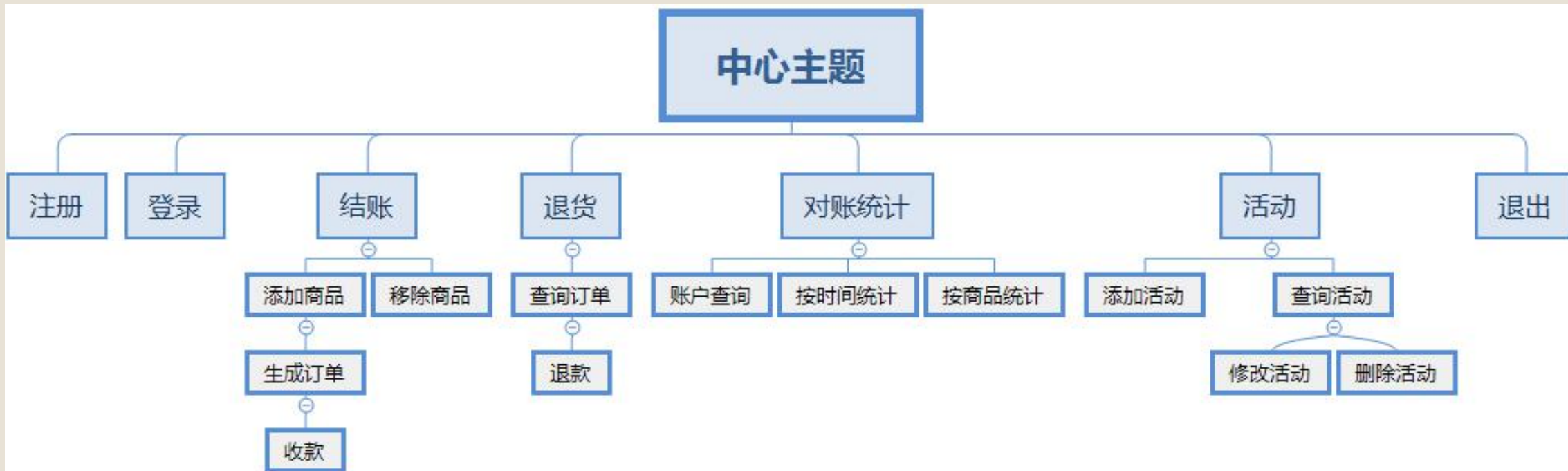
R A R T 0 1

项 目 概 览

Project overview



超市收银系统



该收银系统分为多个功能模块：

首先是一级菜单，分别是注册，登录，退出；

登录后分为二级菜单具有的功能有结账，退货，对账，活动；

然后结账下还有添加商品和移除商品功能，对账有账户查询等功能，活动有活动查询，修改活动等功。

代码编写思路

Code writing ideas



绘制时序图



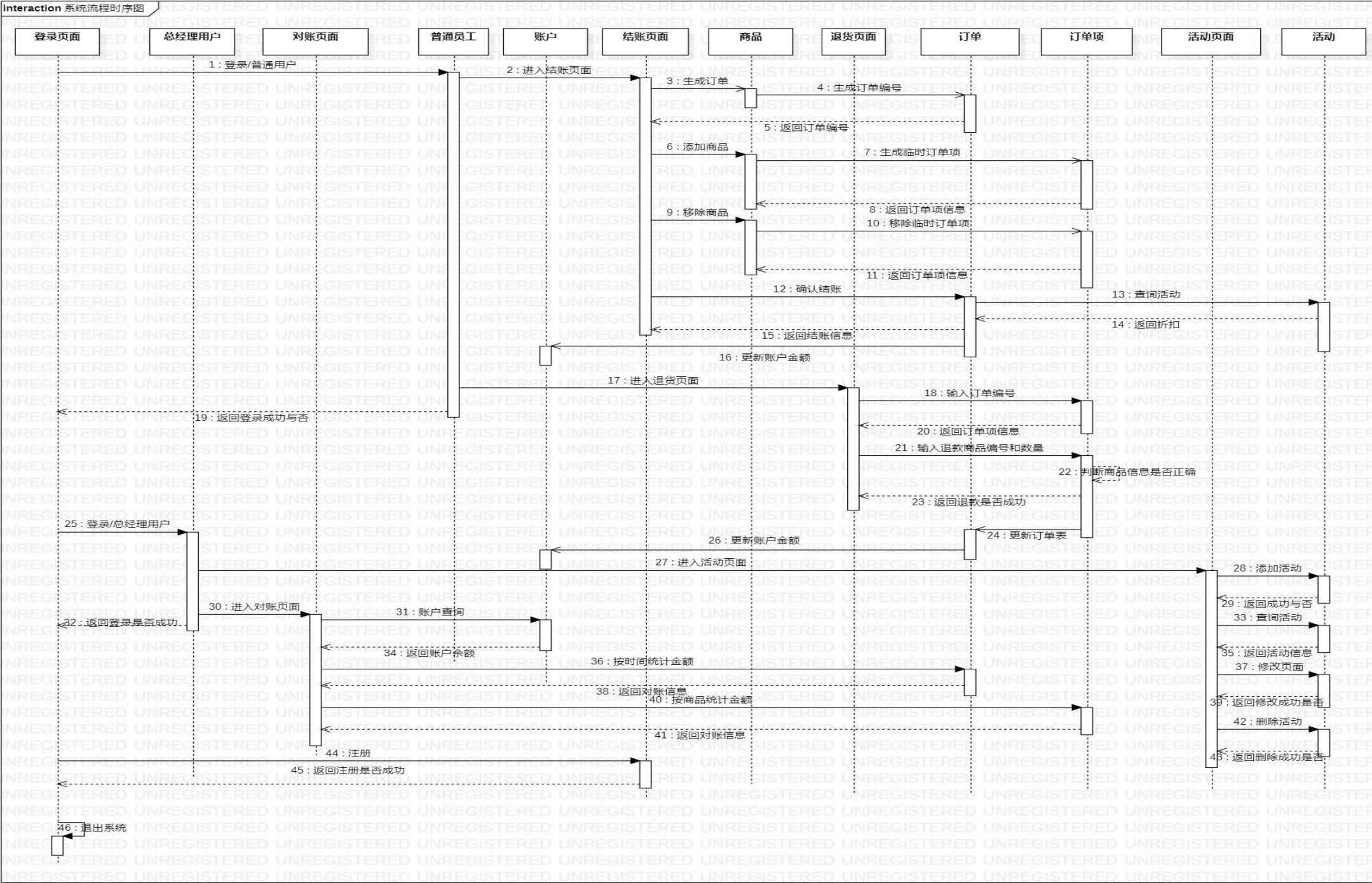
设计数据表



设计接口



时序图



数据表

tb_order /* 订单表 */		
orderTotal	/* 订单总价 */	double(6,2)
orderTrueTotal	/* 订单实际价格 */	double(6,2)
orderDateTime	/* 订单生成日期 */	datetime
orderId	/* 订单编号 */	int

tb_product /* 商品表 */		
productName	/* 商品名称 */	varchar(20)
productPrice	/* 商品单价 */	double(6,2)
productId	/* 商品编号 */	int

tb_orderitem /* 订单项表 */		
orderId	/* 订单编号外键 */	int
productId	/* 商品编号外键 */	int
productName	/* 商品名称 */	varchar(20)
productCount	/* 商品购买数量 */	int
productPrice	/* 商品单价 */	double(6,2)
productTotal	/* 商品总价 */	double(6,2)

tb_activity /* 活动表 */		
activityName	/* 活动名称 */	varchar(50)
criticalTotal	/* 折扣临界金额 */	double(6,2)
startDate	/* 活动开始日期 */	date
endDate	/* 结束日期 */	date
discount	/* 活动折扣 */	double(4,2)
activityId	/* 活动标识id */	int

tb_empuser /* 员工用户表登录收银系统 */		
empUserName	/* 员工用户姓名 */	varchar(20)
empUserDuty	/* 员工用户职位 */	varchar(20)
empUserPwd	/* 员工用户登录密码 */	varchar(20)
empUserId	/* 员工用户id */	int

tb_account /* 收银账户表 */		
accountPwd	/* 收银账户密码 */	varchar(20)
money	/* 收银总金额 */	double(8,2)
accountId	/* 账户id */	int

接口

IAccountDao

IActivityDao

IEmpUserDao

IOrderDao

IOrderItemDao

IProductDao

IAccountService

IActivityService

IEmpUserService

IOrderItemService

IOrderService

IProductService

```
/**
 * Description: selectOrderItem 根据订单编号和商品编号查询唯一订单项
 * 查询请使用BaseDao.selectInfo()方法,
 * @return com.myhd.pojo.OrderItem 返回查询到的订单项
 * @param orderId 订单编号
 * @param productId 商品编号
 * @author huian
 * @Date 2023/8/18
 * */
2 usages 1 implementation Jinhui-Huang
OrderItem selectOrderItem(Integer orderId, Integer productId);
```

```
/**
 * Description: selectAllOrderItem 查询当前订单编号下的全部订单项信息
 * @return java.util.List<com.myhd.pojo.OrderItem>
 * @param orderId 订单编号
 * @author huian
 * @Date 2023/8/19
 * */
2 usages 1 implementation Jinhui-Huang
List<OrderItem> selectAllOrderItem(Integer orderId);
```

```
/**
 * Description: insertOrderDao 应先传入一个只含有生成时间的order对象, 来先生成一个订单id,
 * 再返回给收银员, 用于订单项的填入,
 * 插入请使用BaseDao.updateInfo()方法,
 * @return java.lang.Integer 返回的订单编号 从200001自增
 * @param order 传进来只含有生成时间的order对象
 * @author huian
 * @Date 2023/8/18
 * */
2 usages 1 implementation Jinhui-Huang
Integer insertOrder(Order order);

/**
 * Description: updateOrder 更新订单表, 必须根据订单id进行更新, 更新内容为传入的order对象,
 * 订单id应为先前在insertOrder(Order order)中返回的订单id, 或者用户进行 退款时传进来的订单id,
 * 更新请使用BaseDao.updateInfo()方法,
 * @return java.lang.Integer 返回更新时对订单表影响的行数, 一般都是为1
 * @param order 用于更新的order对象
 * @author huian
 * @Date 2023/8/18
 * */
2 usages 1 implementation Jinhui-Huang
Integer updateOrder(Order order);
```

代码难点



难点

多接口方法调用实现功能

```
/**
 * @description: 退货功能, 应该把已经确认的退货订单项信息集合(必须含有订单编号)传入退货方法里, 再使用订单项
 * 服务层的更新订单项方法, 统计订单项金额, 然后调用订单更新功能更新订单表,
 * @param order 退货后需要更新的订单表对象
 * @param orderItems 确认好退款的订单项集合
 * @return: com.myhd.pojo.Order 返回一个退货后的订单表
 * @author CYQH
 * @date: 2023/08/20 18:56
 */
2 usages  yuexinliang
@Override
public Order returnOfGoods(Order order, List<OrderItem> orderItems) {
    Double total = 0.0;
    Double trueTotal = 0.0;
    Iterator<OrderItem> oldItems = orderItems.iterator();
    while (oldItems.hasNext()){
        OrderItem next = oldItems.next();
        ois.updateOrderItem(next);
        total += next.getProductTotal();
    }
    Double discount = as.selectDiscount(order.getOrderDateTime(),total);
    trueTotal = BigDecimal.valueOf(discount * total).setScale( newScale: 2, RoundingMode.HAL
ias.updateAccount( money: trueTotal-order.getOrderTrueTotal());
    Order order1 = new Order(order.getOrderId(),total,trueTotal,order.getOrderDateTime());
    od.updateOrder(order1);
    return order1;
}
```

主方法逻辑判断

```
/**
 * @description: 结账操作, 内包含添加商品和移除商品, 最终结账
 * @return: void
 * @author CYQH
 * @date: 2023/08/21 8:52
 */
2 usages  yuexinliang *
public static void SettleAccounts(Order order){
    System.out.println("*****结账界面*****");
    HashMap<ArrayList<Integer>, OrderItem> hashMap = new HashMap<>();
    System.out.println("您的订单编号为: "+order.getOrderId());
    try {
        while (true){
            System.out.println("1.添加商品; 2.移除商品; 0.返回上一级菜单");
            System.out.print("请选择您的操作序号: ");
            String str = input.next();
            if ("1".equals(str)){
                break;
            }else if ("2".equals(str)){
                System.out.println("您的订单中暂无商品");
            }else if ("0".equals(str)){
                enterStaffMenu();
            }else {
                System.out.println("操作序号错误, 请重新输入");
            }
        }
    }
}
```

R A R T 0 4

项 目 展 示



感谢您的观看

总结工作 / 计划未来

There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration some form, by injected humour, or randomThere



20XX

● 演讲人：稻小壳 | 公司：金山办公软件有限公司 | 团队：稻壳团队 ●