

第四周课设报告

项目需求描述

客户端

登录

- 输入账号和密码，， 点击登录按钮，进入客户端主页。
- 点击“注册账号”文字，跳转到注册页面。

注册

- 输入用户名和密码，点击注册按钮完成账号注册。
- 点击“返回”文字，跳转到登录界面。

导航栏

- 显示用户的名称，头像和邮箱。
- 带有搜索框，可以迅速搜索需要切换的页面。
- 显示可切换的页面，点击后即可切换至对应页面。

账号管理

- 输入原密码、新密码、再次输入新密码完成对当前用户密码的修改。

编辑个人信息

- 显示用户的个人信息，包括用户名、头像、邮箱、性别、电话号码。
- 可修改头像、邮箱、性别、电话号码等信息，点击修改按钮即可完成修改。

编辑收货地址信息

- 显示用户的所有地址信息，每条地址信息包括收货人姓名、手机号、详细地址。
- 有一个默认地址，可以通过点击默认按钮将当前地址设置为默认地址。
- 展示顺序为默认地址在前，其余地址按创建时间升序排序。
- 可以勾选收货地址进行删除。
- 可以点击添加收货地址按钮添加地址信息。
- 可以点击修改按钮修改当前地址信息。

主页

- 具有海报展示区域，滚动展示一幅幅海报。
- 具有热销商品推荐区域，推荐销量前十的商品。
- 商品信息显示名称、描述、图片、价格、销量。

- 可选择对应商品数量，并将其加入进购物车。

商品搜索页

- 默认展示随机十个商品信息。
- 商品信息显示名称、描述、图片、价格、销量。
- 可选择对应商品数量，并将其加入进购物车。
- 可再随机展示十个商品信息。
- 使用搜索框模糊搜索，并显示搜索结果。

购物车

- 显示用户当前购物车中的所有商品。
- 商品信息显示名称、描述、图片、价格、数量、折扣。
- 显示可用的满减策略。
- 显示商品当前价格与加入购物车前的价格对比。
- 可修改商品数量。
- 可勾选需要结算的商品。
- 可删除购物车中的商品。
- 点击结算按钮，跳转到历史订单页面，清空购物车中对应的商品。

历史订单

- 显示用户所有订单信息。
- 显示每个订单中的所有商品（名称、图片、价格、数量）。

- 显示满减策略。
- 显示总价（原价+折扣后的价格）。
- 点击支付按钮完成对应订单支付。
- 点击取消支付按钮删除未支付订单。
- 点击退款按钮取消已支付的订单。

客服中心

- 显示与客服的对话记录。
- 实时发送接收聊天信息。

日间模式/夜间模式切换

- 点击客户端右上角图标切换日间模式/夜间模式。

服务端

服务器控制

- 点击开启服务器按钮，完成服务器启动。
- 点击关闭服务器按钮，完成服务器关闭。

显示单个商品信息表格

- 显示每个商品名称、描述、图片、价格、销量。
- 以表格形式展示。

相关技术

C/S框架

C/S（客户端/服务器）架构是一种分布式系统模型，其中客户端负责与用户交互，发送请求并显示结果，而服务器负责处理请求、管理数据和业务逻辑。客户端和服务器通过网络通信，常见的协议有HTTP和TCP/IP。C/S架构的优点是功能分离、效率高、安全性好，但也存在客户端依赖性强、扩展性受限等缺点。

持久层

本项目的持久层主要使用了Qt提供的数据库操作类QSqlDatabase和QSqlQuery。

QSqlDatabase将程序与数据库建立连接，执行数据库管理操作。通过QSqlDatabase，可以打开、关闭数据库，并管理多个数据库连接。它支持多种数据库类型，如SQLite、MySQL、PostgreSQL等，提供统一的接口来访问不同数据库。

QSqlQuery是用于执行SQL查询的类，可以通过它发送SQL语句（如SELECT、INSERT、UPDATE等）到数据库，并获取结果。QSqlQuery 提供了处理查询结果的方法，比如next()、value()等，用于遍历和读取查询返回的记录。它与

QSqlDatabase配合使用，通常在数据库连接建立后执行SQL操作

异步通信

本项目的通信协议采用TCP/IP协议，并使用Qt提供的QTcpSocket类和QTcpServer类实现异步通信。

QTcpServer 是 Qt提供的用于监听和接受TCP网络连接的类。它可以在指定的端口上等待客户端的连接请求，当有客户端连接时，QTcpServer会创建一个新的QTcpSocket实例来与客户端进行通信。通过QTcpServer，可以实现服务器端的网络监听和处理多个客户端的连接。

QTcpSocket是Qt中用于处理TCP网络通信的类，提供了客户端与服务器端之间的数据交换功能。通过 QTcpSocket，客户端可以发送和接收数据，处理连接的建立、断开以及数据的读写等操作。QTcpSocket 支持异步操作，可以通过信号与槽机制处理网络事件，如连接成功、数据接收等。

线程池

本项目的线程池的实现参考了开源项目[ThreadPool](#)。简单来说，线程池使用thread管理多个工作线程，通过任务队列分配任务，通过条件变量实现线程同步，通过mutex和unique_lock保护共享资源，并使用packaged_task和future

实现任务结果的异步返回。实现了任务的异步执行，在保证线程安全的条件下，提高了程序的运行效率。

Qt信号槽机制

本项目没有使用观察者模式和订阅模式等设计模式，而是使用了效果更为出色的Qt信号槽机制，实现了模块间的通信。

Qt 的信号槽机制是一种对象间通信的方式，用于处理事件驱动的编程模型。对象通过发射信号（Signal）来通知其他对象某个事件发生，而接收信号的对象通过槽（Slot）来响应这些事件。当信号发射时，所有连接到该信号的槽函数会被自动调用。信号和槽不需要显式的函数调用，它们通过 Qt 的事件循环和对象间的连接来实现通信。信号槽机制使得 Qt 中的对象解耦，提高了代码的模块化和可维护性。

前端和后端概要设计

模块的划分

持久层

使用了一系列mapper类实现数据库的相关增删改查操作，供业务层调用。

业务层

使用了一系列service类实现不同功能模块的业务逻辑，并进行了相应的异常处理。

控制层

服务端使用TcpServer类统一处理客户端的通信请求。
客户端的通信逻辑内嵌在MainWindow类中，并通过Qt的信号槽机制实现通信。

UI设计

使用对Qt组件进行一定程度封装的仿Windows的Fluent Design风格的开源Qt组件库——[ElaWidgetTools](#)辅助图形界面设计。

主要类的划分

- 用户模块
 - UserMapper
 - UserService
- 地址模块
 - AddressMapper
 - AddressService
- 商品模块
 - ProductMapper
 - ProductService

- 购物车模块
 - CartMapper
 - CartService
- 订单模块
 - OrderMapper
 - OrderItemMapper
 - OrderService
- 聊天模块
 - ChatRecordMapper
 - ChatRecordService
- 折扣模块
 - DiscountMapper
- 满减策略模块
 - StrategyMapper
- 网络通信相关
 - TcpServer
 - JsonResult
 - JsonResult
- 业务层异常相关类
 - ServiceError
 - UserNameDuplicatedError
 - InsertError
 - UserNotFoundError

- PasswordNotMatchError
- UpdateError
- AddressCountError
- AccessDeniedError
- AddressNotFoundError
- RemoveError
- ProductNotFoundError
- InvalidArgumentError
- CartNotFoundError
- ProductNotEnoughError
- NoProductInCartError
- OrderNotFoundError

数据库设计

t_user (用户信息)

字段名称	数据类型	描述
uid	INT	用户id
user_name	VARCHAR(20)	用户名
password	CHAR(32)	密码
salt	CHAR(36)	盐值
phone_number	VARCHAR(20)	电话号码
email	VARCHAR(20)	电子邮箱
gender	INT	性别 (0-女 1-男)
avatar	VARCHAR(50)	头像
is_delete	INT	是否删除 (0-未删除 1-已删除)
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_address (地址信息)

字段名称	数据类型	描述
aid	INT	收货地址id
uid	INT	归属的用户id
name	VARCHAR(20)	收货人姓名
address	VARCHAR(60)	详细地址
phone_number	VARCHAR(20)	手机号
is_default	INT	是否为默认地址 (0-不默认 1-默认)
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_product (商品信息)

字段名称	数据类型	描述
pid	INT	商品id
cid	INT	商品类别id
name	VARCHAR(100)	商品名称
price	DECIMAL(10, 2)	商品价格
description	VARCHAR(255)	商品描述
num	INT	库存
image	VARCHAR(255)	图片路径
status	INT	商品状态(1: 上架 2: 下架 3:删除)
priority	INT	显示优先级
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_cart (购物车)

字段名称	数据类型	描述
cid	INT	购物车id
uid	INT	用户id
pid	INT	商品id
initial_price	DECIMAL(10, 2)	加入购物车时的价格
num	INT	商品数量
is_choiced	INT	是否被勾选
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_discount (折扣)

字段名称	数据类型	描述
did	INT	折扣策略id
pid	INT	商品id
detail	INT	具体折扣（例：80八折）
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_strategy (满减策略)

字段名称	数据类型	描述
sid	INT	满减策略id
target	INT	要触发折扣应达到的订单总额
saved_money	INT	优惠的元数
is_valid	INT	折扣是否有效
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_order (订单)

字段名称	数据类型	描述
oid	INT	订单id
uid	INT	用户id
receive_name	VARCHAR(20)	收货人姓名
receive_address	VARCHAR(60)	收货人详细地址
receive_phone_number	VARCHAR(20)	收货人手机号
total_price	DECIMAL(10, 2)	总价
state	INT	状态 0-未支付 1-已支付 2-已取消 3-已完成 4-已退款
order_time	DATETIME	下单时间
pay_time	DATETIME	支付时间
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_order_item (订单中的商品)

字段名称	数据类型	描述
id	INT	记录id
oid	INT	所归属的订单id
pid	INT	商品id
num	INT	购买数量
price	DECIMAL(10, 2)	商品单价（已经过折扣）
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

t_chat_record (聊天记录)

字段名称	数据类型	描述
id	INT	记录id
sender_id	INT	发送者id
receiver_id	INT	接收者id
message	VARCHAR(1000)	聊天信息
created_user	VARCHAR(20)	创建者
created_time	DATETIME	创建时间
modified_user	VARCHAR(20)	最后修改者
modified_time	DATETIME	最后修改时间

通信协议设计

请求码

名称	取值	描述
Regisiter	0	用户注册请求
Login	1	用户登录请求
ChangePassword	2	修改密码请求
GetUserInfo	3	获取用户信息请求
ChangeUserInfo	4	修改用户信息请求
AddAddress	5	添加地址请求
GetAddressInfo	6	获取地址信息请求
ChangeDefaultAddress	7	修改默认地址请求
RemoveAddress	8	删除地址请求
GetTopProductInfo	9	获取热门商品信息请求
GetProductInfo	10	获取商品详细信息请求
AddCart	11	添加商品到购物车请求
ChangeCartNum	12	修改购物车中商品数量请求
GetCartInfo	13	获取购物车信息请求
RemoveCart	14	从购物车移除商品请求

名称	取值	描述
ChangeCartIsChoiced	15	修改购物车中商品选择状态请求
AddOrder	16	添加订单请求
PayOrder	17	支付订单请求
CancelOrder	18	取消订单请求
RefundOrder	19	退款订单请求
GetOrderInfo	20	获取订单信息请求
GetOrderItemInfo	21	获取订单项信息请求

响应码

名称	取值	描述
UnknownError	0	未知错误
RegisterDone	1	注册成功
RegisterUserNameDuplicated	2	用户名重复
LoginDone	3	登录成功
LoginUserNotFound	4	用户不存在
LoginPasswordNotMatch	5	密码不匹配
ChangePasswordDone	6	修改密码成功

名称	取值	描述
ChangePasswordUserNotFound	7	用户不存在
ChangePasswordPasswordNotMatch	8	原密码不匹配
GetUserInfoDone	9	获取用户信息成功
GetUserInfoUserNotFound	10	用户不存在
ChangeUserInfoDone	11	修改用户信息成功
ChangeUserInfoUserNotFound	12	用户不存在

名称	取值	描述
AddAddressDone	13	添加地址成功
AddAddressAddressCountLimit	14	地址数量达到上限
GetAddressInfoDone	15	获取地址信息成功
ChangeDefaultAddressDone	16	修改默认地址成功
ChangeDefaultAddressAccessDenied	17	无权限修改默认地址

名称	取值	描述
ChangeDefaultAddressAddressNotFound	18	默认地址未找到
RemoveAddressDone	19	删除地址成功
RemoveAddressAddressNotFound	20	地址未找到
RemoveAddressAccessDenied	21	无权限删除地址
GetTopProductInfoDone	22	获取热门商品信息成功

名称	取值	描述
GetProductInfoDone	23	获取商品信息成功
AddCartDone	24	添加购物车成功
AddCartUserNotFound	25	用户不存在
AddCartProductNotFound	26	商品未找到
ChangeCartNumDone	27	修改购物车商品数量成功

名称	取值	描述
ChangeCartNumCartNotFound	28	购物车中商品未找到
GetCartInfoDone	29	获取购物车信息成功
RemoveCartDone	30	从购物车移除商品成功
RemoveCartCartNotFound	31	购物车中商品未找到

名称	取值	描述
ChangeCartIsChoicedDone	32	修改购物车商品选择状态成功
ChangeCartIsChoicedCartNotFound	33	购物车中商品未找到
AddOrderDone	34	添加订单成功
AddOrderUserNotFound	35	用户不存在
AddOrderAddressNotFound	36	地址未找到

名称	取值	描述
AddOrderAccessDenied	37	无权限添加订单
AddOrderNoProductInCart	38	购物车无商品
AddOrderProductNotEnough	39	商品库存不足
PayOrderDone	40	支付订单成功
PayOrderOrderNotFound	41	订单未找到
PayOrderAccessDenied	42	无权限支付订单

名称	取值	描述
CancelOrderDone	43	取消订单成功
CancelOrderOrderNotFound	44	订单未找到
CancelOrderAccessDenied	45	无权限取消订单
RefundOrderDone	46	退款订单成功
RefundOrderOrderNotFound	47	订单未找到
RefundOrderAccessDenied	48	无权限退款订单

名称	取值	描述
GetOrderInfoDone	49	获取 订单 信息 成功
GetOrderItemInfoDone	50	获取 订单 项信 息成 功