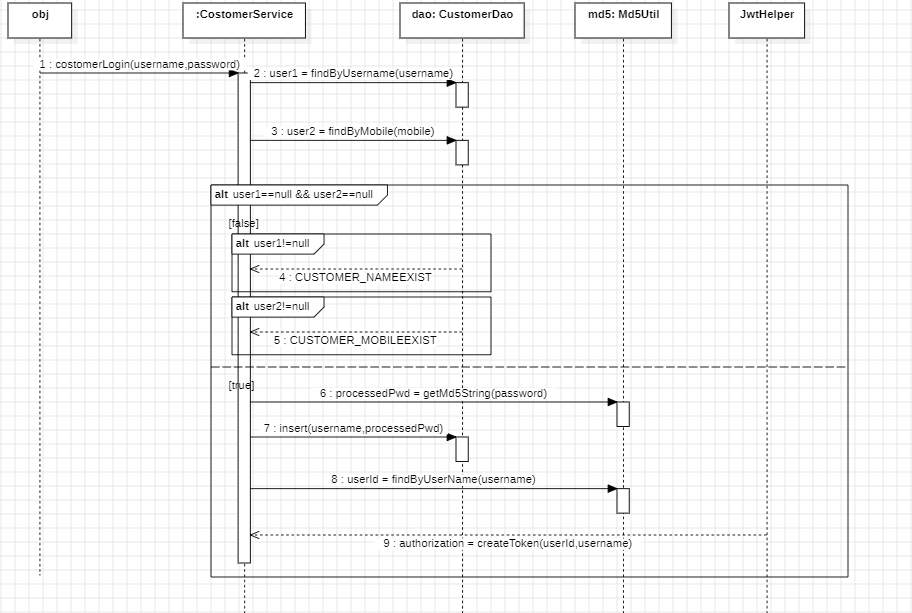
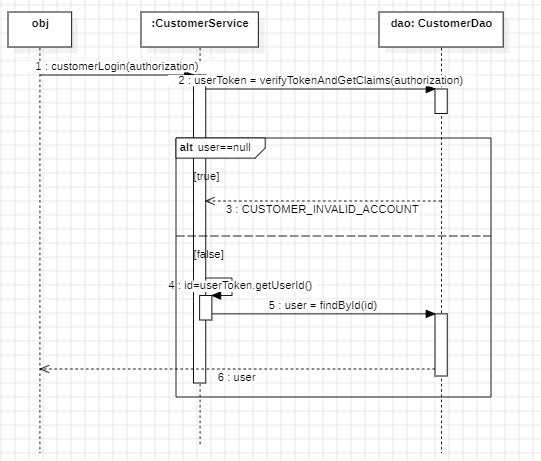
# 3程序描述

## 3.6程序逻辑

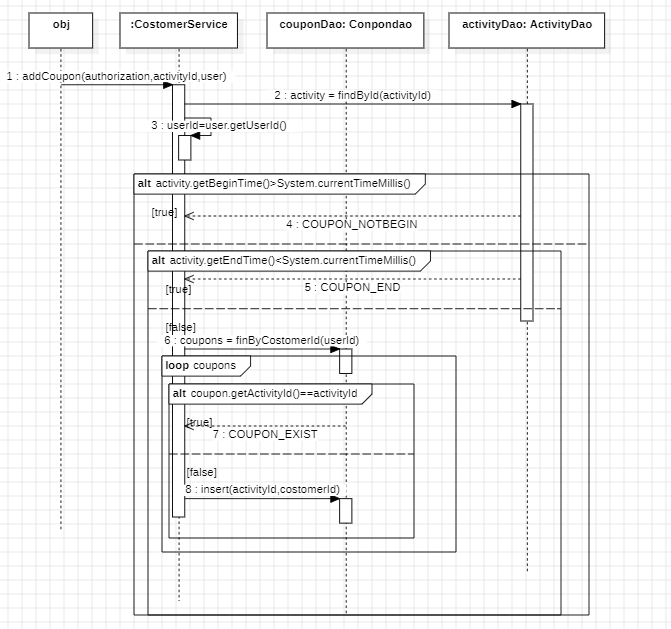
1.顾客注册



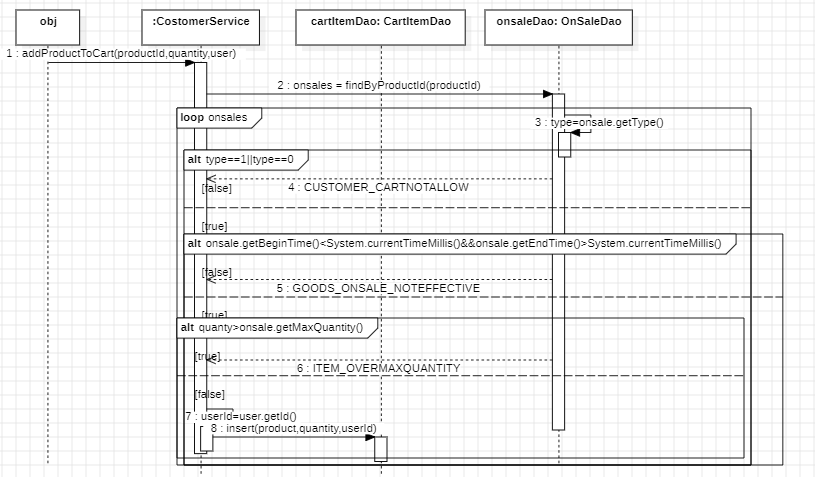
2.顾客登录



3.顾客领取优惠券



4.购物车新增商品

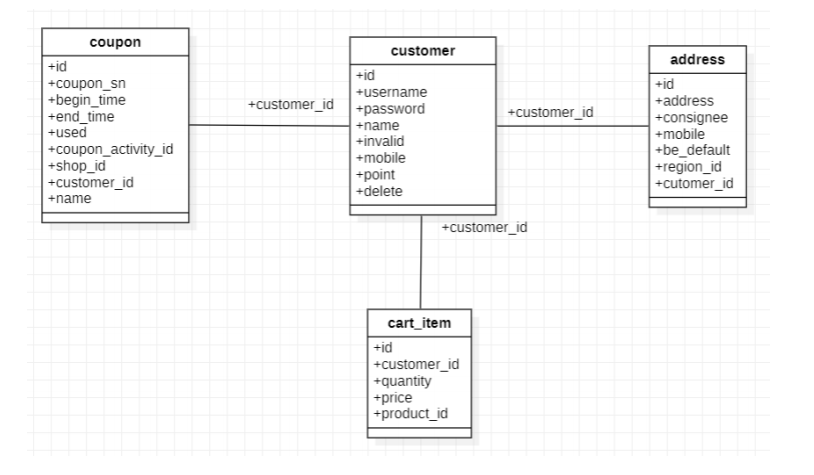


## 3.7接口

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | 传入参数 | 功能 | 返回参数 | API种类 |
| /customers | body | 注册用户 | errno  errmsg  data | POST |
| /customers | authorization | 买家查看自己信息 | errno  errmsg  data | GET |
| /customers | body | 买家修改自己的信息 | errno  errmsg | PUT |
| /customers/password | body | 用户修改密码 | errno  errmsg | PUT |
| /customers/password/reset | body | 用户重置密码 | errno  errmsg | PUT |
| /customers/login | body | 用户名密码登录 | errno  errmsg  data | POST |
| /customers/logout | authorization | 用户登出 | errno  errmsg  data | GET |
| /carts | authorization  page  pagesize | 买家获得购物车列表 | errno  errmsg  data | GET |
| /carts | authorization  body | 买家将商品加入购物车 | errno  errmsg  data | POST |
| /carts | authorization | 买家清空购物车 | errno  errmsg | DELETE |
| /addresses | authorization  body | 买家新增地址 | errno  errmsg  data | POST |
| /addresses | authorization  id  body | 买家修改自己的地址信息 | errno  errmsg | PUT |

## 3.8存储分配

顾客模块



1.优惠券表coupon

记录顾客所拥有的优惠券

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 物理表名 | coupon | | | | | |
| 主键 | id | | | | | |
| 数据源类型 | Mysql | | | | | |
| 业务主键 | 无 | | | | | |
| 字段列表 | | | | | | |
| 中文名称 | 列名 | 数据类型 | 默认 | 主键 | 非空 | 外键 |
| Id | id | Bigint |  | √ | √ |  |
| 优惠卷编号 | coupon\_sn | Bigint |  |  | √ |  |
| 创建时间 | begin\_time | Date |  |  | √ |  |
| 结束时间 | end\_time | Date |  |  | √ |  |
| 是否被使用 | used | Tinyint |  |  | √ |  |
| 活动id | coupon\_activity\_id | Bigint |  |  |  | √ |
| 商铺id | shop\_id | Bigint |  |  |  | √ |
| 顾客id | customer\_id | Bigint |  |  |  | √ |
| 优惠卷名称 | name | Varchar(255) |  |  | √ |  |

2.顾客表customer

记录顾客的个人信息

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 物理表名 | customer | | | | | |
| 主键 | id | | | | | |
| 数据源类型 | Mysql | | | | | |
| 业务主键 | 无 | | | | | |
| 字段列表 | | | | | | |
| 中文名称 | 列名 | 数据类型 | 默认 | 主键 | 非空 | 外键 |
| id | id | Bigint |  | √ | √ |  |
| 用户名 | username | Varchar(255) |  |  | √ |  |
| 密码 | password | Varchar(255) |  |  | √ |  |
| 真实姓名 | name | Varchar(255) |  |  | √ |  |
| 是否被禁用 | invalid | Tinyint |  |  | √ |  |
| 电话号码 | mobile | Bigint |  |  | √ |  |
| 积点 | point | Bigint |  |  |  |  |
| 是否被注销 | delete | Tinyint |  |  |  |  |

3.地址表address

记录顾客所有的地址

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 物理表名 | address | | | | | |
| 主键 | Id | | | | | |
| 数据源类型 | Mysql | | | | | |
| 业务主键 | 无 | | | | | |
| 字段列表 | | | | | | |
| 中文名称 | 列名 | 数据类型 | 默认 | 主键 | 非空 | 外键 |
| id | Id | Bigint |  | √ | √ |  |
| 地址 | address | Varchar(255) |  |  | √ |  |
| 收货人 | consignee | Varchar(255) |  |  | √ |  |
| 电话号码 | mobile | Varchar(255) |  |  | √ |  |
| 是否是默认地址 | be\_default | Tinyint |  |  | √ |  |
| 地区id | region\_id | Bigint |  |  |  | √ |
| 顾客id | customer\_id | Bigint |  |  |  | √ |

4.购物车表cart\_item

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 物理表名 | cart\_item | | | | | |
| 主键 | Id | | | | | |
| 数据源类型 | Mysql | | | | | |
| 业务主键 | 无 | | | | | |
| 字段列表 | | | | | | |
| 中文名称 | 列名 | 数据类型 | 默认 | 主键 | 非空 | 外键 |
| id | Id | Bigint |  | √ | √ |  |
| 顾客id | costomer\_id | Bigint |  |  |  | √ |
| 数量 | quantity | Bigint |  |  | √ |  |
| 价格 | price | Double |  |  | √ |  |
| 商品id | product\_id | Bigint |  |  |  | √ |

## 3.9限制条件

1. 经费限制：平台的开发需要一定的经费支持，包括购买服务器、支付软件使用费等，项目总经费为华为代金券共 300 元。本项目服务器使用华为云服务器，团队服务器数 量未定。

2. 开发时间限制：平台开发时间限制为一学期，即 2023 年到 2024 年第一学期。团队 需要在规定的时间内完成开发并交付。

3. 硬件限制：服务器的硬件资源有限，性能与服务器数量未定。

4. 软件限制：采用 Java spring-boot 技术栈进行开发。

5. 安全限制：项目必须保证系统和数据的安全性，防止潜在风险和威胁。

6. 法律和合规限制：项目必须符合适用的法律法规和相关行业标准，遵循合规要求。

## 3.10测试要点

1. 功能测试：验证软件系统的各项功能是否满足预期，包括输入输出、界面操作、业务逻辑等方面的测试。

2. 性能测试：评估软件系统在不同负载条件下的性能表现，包括响应时间、吞吐量、稳定性等指标的测试，以适应电子商城系统高并发大负载的运行需求。

3. 安全测试：检查软件系统的安全性，包括身份验证、数据加密、访问控制等方面的测试。从而保护服务器免收不法分子攻击，防止用户的信息泄露造成经济损失等等。

4. 兼容性测试：验证软件系统在不同平台、操作系统、浏览器或设备上的兼容性，确保系统在各种环境下正常运行。

5. 接口测试：测试软件系统与其他系统或组件之间的接口功能和数据传输的正确性。

6. 异常处理测试：测试软件系统对异常情况的处理能力，如错误输入、网络中断、数据库故障等。

7. 回归测试：在软件系统进行修改或更新后，对已测试过的功能进行再次测试，以确保新修改没有引入新的错误。

8. 自动化测试：利用自动化测试工具和脚本执行一部分或全部的测试用例，提高测试效率和覆盖范围。目前已经实现：系统每天0:00 6:00 12:00 18:00会自动进行一次单元测试，

系统每天00:50 4:50 8:50 12:50 16:50 20:50会自动进行一次集成测试，自动生成output.log文件，可以看到测试时的完整输出。