《工程实践4（软件产品设计）》

详细设计说明书

|  |  |
| --- | --- |
| 题 目： | 网上花店销售系统 |
| 组 长： | 陈建文2019082070 |
| 成 员： | 陈 成2019101233 |
| 李华祥2021081167 |
| 所在学院： | 软件工程学院 |
| 年级班级： | 软工215班 |
| 指导教师： | 王 燮 |

2024年4月

成都信息工程大学 软件工程学院

目录

[1. 引言 7](#_Toc168064662)

[1.1. 编写目的 7](#_Toc168064663)

[1.2. 背景 7](#_Toc168064664)

[2. 软件结构 9](#_Toc168064665)

[3. 系统模块设计说明 11](#_Toc168064666)

[3.1. 用户模块 11](#_Toc168064667)

[3.1.1. 系统流程 11](#_Toc168064668)

[3.1.1.1. 用户注册与登录 11](#_Toc168064669)

[3.1.1.2. 个人信息管理 13](#_Toc168064670)

[3.1.1.3. 用户行为记录与分析 14](#_Toc168064671)

[3.1.1.4. 安全与权限管理 14](#_Toc168064672)

[3.1.2. 系统管理模块类和接口设计 14](#_Toc168064673)

[3.1.2.1. Controller层 16](#_Toc168064674)

[3.1.2.1.1. HelloController类 16](#_Toc168064675)

[3.1.2.1.2. ChatController类 16](#_Toc168064676)

[3.1.2.1.3. GetSessionController类 16](#_Toc168064677)

[3.1.2.1.4. LoginController类 17](#_Toc168064678)

[3.1.2.1.5. LogoutController类 17](#_Toc168064679)

[3.1.2.1.6. RegisterController类 18](#_Toc168064680)

[3.1.2.1.7. TUserContactController类 18](#_Toc168064681)

[3.1.2.1.8. TUserController类 19](#_Toc168064682)

[3.1.2.1.9. TUserLoginBehaviorController类 21](#_Toc168064683)

[3.1.2.1.10. TUserSearchBehaviorController类 22](#_Toc168064684)

[3.1.2.1.11. TUserSettingsController类 23](#_Toc168064685)

[3.1.2.2. Service层 24](#_Toc168064686)

[3.1.2.2.1. ChatService接口 24](#_Toc168064687)

[3.1.2.2.2. EmailService接口 25](#_Toc168064688)

[3.1.2.2.3. GetHexService接口 25](#_Toc168064689)

[3.1.2.2.4. LoginService接口 26](#_Toc168064690)

[3.1.2.2.5. RegisterService接口 26](#_Toc168064691)

[3.1.2.2.6. TUserContactService接口 27](#_Toc168064692)

[3.1.2.2.7. TUserLoginBehaviorService接口 27](#_Toc168064693)

[3.1.2.2.8. TUserSearchBehaviorService接口 27](#_Toc168064694)

[3.1.2.2.9. TUserService接口 28](#_Toc168064695)

[3.1.2.2.10. TUserSettingsService接口 28](#_Toc168064696)

[3.1.2.3. Dao层 28](#_Toc168064697)

[3.1.2.3.1. TUserContactDao类 29](#_Toc168064698)

[3.1.2.3.2. TUserDao类 30](#_Toc168064699)

[3.1.2.3.3. TUserLoginBehaviorDao类 31](#_Toc168064700)

[3.1.2.3.4. TUserSearchBehaviorDao类 33](#_Toc168064701)

[3.1.2.3.5. TUserSettingsDao类 35](#_Toc168064702)

[3.1.2.4. Entity层 37](#_Toc168064703)

[3.1.2.4.1. Tuser类 37](#_Toc168064704)

[3.1.2.4.2. TUserContact类 37](#_Toc168064705)

[3.1.2.4.3. TUserLoginBehavior类 38](#_Toc168064706)

[3.1.2.4.4. TUserSearchBehavior类 38](#_Toc168064707)

[3.1.2.4.5. TUserSettings类 39](#_Toc168064708)

[3.1.2.5. Interceptor 39](#_Toc168064709)

[3.1.2.5.1. LoginConfig类 39](#_Toc168064710)

[3.1.2.5.2. UserLoginInterceptor类 40](#_Toc168064711)

[3.2. 商品模块 40](#_Toc168064712)

[3.2.1. 系统流程 40](#_Toc168064713)

[3.2.1.1. 公告模块设计说明 40](#_Toc168064714)

[3.2.1.2. 分类模块设计说明 42](#_Toc168064715)

[3.2.1.3. 商品模块设计说明 43](#_Toc168064716)

[3.2.2. 系统管理模块类和接口设计 45](#_Toc168064717)

[3.2.2.1. TFlowerClassifyController类 45](#_Toc168064718)

[3.2.2.2. TFlowerController类 47](#_Toc168064719)

[3.2.2.3. TPublicNoticeController类 48](#_Toc168064720)

[3.2.2.4. TFlowerClassify类 49](#_Toc168064721)

[3.2.2.5. TFlower类 50](#_Toc168064722)

[3.3. 订单模块 50](#_Toc168064723)

[3.3.1. 系统流程 50](#_Toc168064724)

[3.3.1.1. 购物车模块设计说明 50](#_Toc168064725)

[3.3.1.2. 订单列表模块设计说明 52](#_Toc168064726)

[3.3.1.3. 订单详情模块设计说明 54](#_Toc168064727)

[3.3.2. 系统管理模块类和接口设计 55](#_Toc168064728)

[3.3.2.1. TCartController类 55](#_Toc168064729)

[3.3.2.2. TorderController类 56](#_Toc168064730)

[3.3.2.3. TOrderDetailsController类 58](#_Toc168064731)

[3.3.2.4. TCart类 59](#_Toc168064732)

[3.3.2.5. TOrder类 60](#_Toc168064733)

[3.3.2.6. TOrderDetails类 60](#_Toc168064734)

1. 引言
   1. 编写目的

本详细设计文档旨在明确鲜花管理系统各个功能模块的设计细节和实现方案，确保系统开发过程中的一致性和规范性。具体目标如下：

明确系统功能：详细描述系统的各个功能模块，包括用户管理、鲜花管理、订单管理等，确保所有团队成员对系统的功能有统一的理解。

技术细节说明：提供每个功能模块的技术实现方案，包括数据库设计、接口定义、业务逻辑等，确保开发过程中的技术一致性。

接口和数据流：定义系统各部分之间的接口和数据流，确保系统各组件之间的协同工作和数据交互的顺畅。

设计规范和标准：明确系统开发的设计规范和标准，确保代码的可维护性、可扩展性和高质量。

风险评估和解决方案：识别系统设计和实现过程中可能遇到的风险，并提供相应的解决方案，以确保系统的稳定性和可靠性。

通过本详细设计文档的编写，旨在为开发团队提供一个全面、系统的设计指南，确保鲜花管理系统开发工作的高效推进和顺利完成。

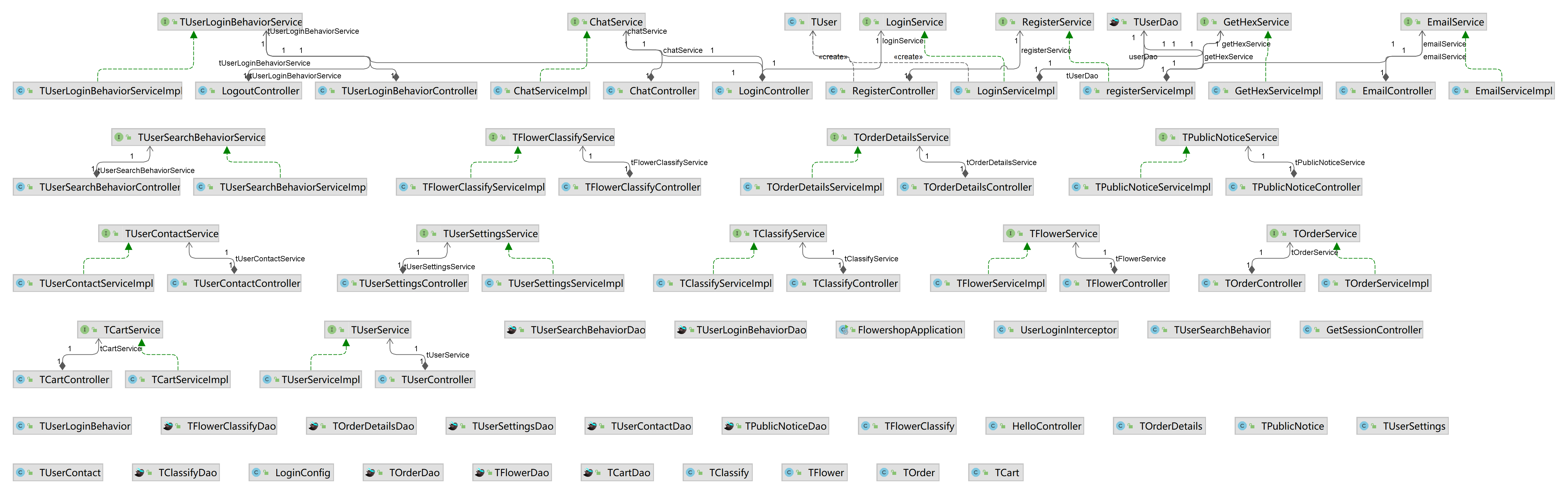
* 1. 背景

随着电子商务的快速发展，鲜花行业也逐渐走向线上化和数字化。鲜花管理系统应运而生，为用户提供了一站式的鲜花购买服务，从鲜花选择、订单管理到配送服务，极大地提升了用户体验。同时，商家可以通过该系统高效地管理库存、分类、订单以及用户信息，从而提升运营效率，减少管理成本。

鲜花管理系统的开发旨在解决传统鲜花销售中的痛点，例如库存管理不善、订单处理效率低、用户需求难以精准满足等问题。通过数字化的解决方案，不仅能为用户提供更便捷的购物体验，还能帮助商家实现智能化管理，提升市场竞争力。

1. 软件结构





1. 系统模块设计说明
   1. 用户模块

用户模块是线上鲜花销售系统中至关重要的一部分，它涵盖了用户管理、个人信息管理以及用户行为分析等多个功能。



图3-1 用户模块划分

* + 1. 系统流程
       1. 用户注册与登录

提供用户注册页面，用户需填写用户名、密码、邮箱等信息完成注册。注册时需验证用户名和邮箱的唯一性，密码需要符合一定的复杂度要求。注册成功后，系统自动为用户分配唯一的用户ID。提供用户登录页面，用户需输入已注册的用户名和密码完成登录。登录时需要验证用户的身份信息，验证通过后允许用户访问系统其他功能，否则提示登录失败。



图3-1-1 登录流程图

* + - 1. 个人信息管理

在个人信息页面，用户可以查看已填写的个人资料，包括用户名、邮箱、联系方式等信息。用户可以编辑个人资料，修改用户名、邮箱、联系方式等信息。修改后需要进行合法性验证，并更新至数据库。

提供添加新收货地址的功能，用户需填写收货人姓名、联系电话、详细地址等信息。添加的地址可以在下单时选择使用。用户可以编辑已添加的收货地址信息，包括修改联系电话、详细地址等内容。用户可以删除不需要的收货地址，删除后系统将不再保存该地址信息。



图3-1-2信息编辑流程图

* + - 1. 用户行为记录与分析

系统记录用户的登录、登出、搜索、浏览商品等行为，包括行为发生时间和具体内容。对用户行为数据进行分析，了解用户的偏好和习惯，为商品推荐、营销活动等提供数据支持。

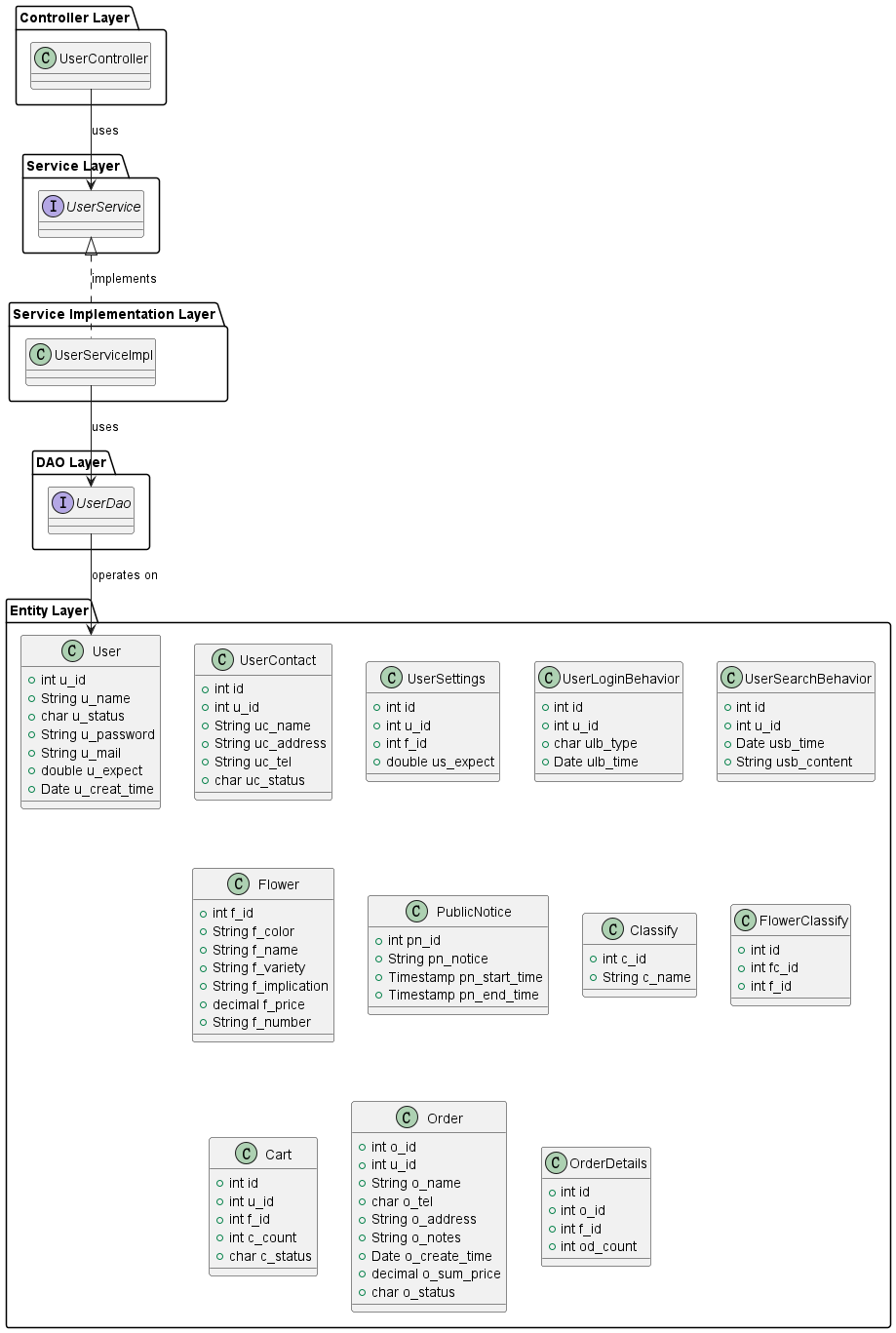


图3-1-3 获取用户信息流程图

* + - 1. 安全与权限管理

系统对用户密码进行加密存储，确保用户信息的安全性。区分用户身份，设置管理员和普通用户权限，管理员可以对用户信息进行管理，普通用户只能访问和修改自己的信息。

* + 1. 系统管理模块类和接口设计



* + - 1. Controller层
         1. HelloController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | HelloController |
| 类功能 | | 展示主页 |
| 方法1 | 名称 | public void hello |
| 功能 | 展示主页 |
| 输入 | HttpServletRequest request,  HttpServletResponse response |
| 输出 | 无 |

* + - * 1. ChatController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | ChatController |
| 类功能 | | 会话 |
| 方法1 | 名称 | public ResponseEntity<String> chat |
| 功能 | 会话 |
| 输入 | String message |
| 输出 | 聊天返回结果 |

* + - * 1. GetSessionController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | GetSessionController |
| 类功能 | | 提供当前登录用户的登录信息 |
| 方法1 | 名称 | public TUser getSession |
| 功能 | 提供当前登录用户的登录信息 |
| 输入 | HttpServletRequest request,  HttpServletResponse response |
| 输出 | 当前登录用户的登录信息 |

* + - * 1. LoginController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | LoginController |
| 类功能 | | 登录 |
| 方法1 | 名称 | public TUser login |
| 功能 | 登录 |
| 输入 | String email，  String password  HttpServletRequest request,  HttpServletResponse response |
| 输出 | TUser |

* + - * 1. LogoutController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | LogoutController |
| 类功能 | | 登出 |
| 方法1 | 名称 | public Integer logout |
| 功能 | 登出 |
| 输入 | HttpServletRequest request, |
| 输出 | 登出结果 |

* + - * 1. RegisterController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | RegisterController |
| 类功能 | | 注册 |
| 方法1 | 名称 | public Integer register |
| 功能 | 注册 |
| 输入 | TUser |
| 输出 | 注册结果 |

* + - * 1. TUserContactController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserContactController |
| 类功能 | | 收货联系方式表;(TUserContact)表控制层 |
| 方法1 | 名称 | public ResponseEntity<Page<TUserContact>>  queryByPage |
| 功能 | 分页查询 |
| 输入 | TUserContact tUserContact,  PageRequest pageRequest |
| 输出 | 查询结果 |
| 方法2 | 名称 | public ResponseEntity<Page<TUserContact>>  queryById |
| 功能 | 主键查询 |
| 输入 | Integer id |
| 输出 | 查询结果 |
| 方法3 | 名称 | public ResponseEntity<Page<TUserContact>>  add |
| 功能 | 新增数据 |
| 输入 | TUserContact tUserContact |
| 输出 | 新增结果 |
| 方法4 | 名称 | public ResponseEntity<Page<TUserContact>>  edit |
| 功能 | 编辑数据 |
| 输入 | TUserContact tUserContact |
| 输出 | 编辑结果 |
| 方法5 | 名称 | public ResponseEntity<Page<TUserContact>>  deleteById |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 删除结果 |

* + - * 1. TUserController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserController |
| 类功能 | | 用户表;(TUser)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询 |
| 输入 | Page<TUser> page,  TUser tUser |
| 输出 | 查询结果 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 主键查询 |
| 输入 | Integer id |
| 输出 | 查询结果 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | TUser tUser |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 编辑数据 |
| 输入 | TUser tUser |
| 输出 | 编辑结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 删除结果 |

* + - * 1. TUserLoginBehaviorController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserLoginController |
| 类功能 | | 用户登录行为记录表;  (TUserLoginBehavior)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询 |
| 输入 | Page<TUser> page,  TUserLoginBehavior tUserLoginBehavior |
| 输出 | 查询结果 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 主键查询 |
| 输入 | Integer id |
| 输出 | 查询结果 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | TUserLoginBehavior tUserLoginBehavior |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 编辑数据 |
| 输入 | TUserLoginBehavior tUserLoginBehavior |
| 输出 | 编辑结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 删除结果 |

* + - * 1. TUserSearchBehaviorController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSearchBehaviorController |
| 类功能 | | 用户搜索行为记录表;  (TUserSearchBehavior)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询 |
| 输入 | Page<TUser> page,  TUserSearchBehavior tUserSearchBehavior |
| 输出 | 查询结果 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 主键查询 |
| 输入 | Integer id |
| 输出 | 查询结果 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | TUserSearchBehavior tUserSearchBehavior |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 编辑数据 |
| 输入 | TUserSearchBehavior tUserSearchBehavior |
| 输出 | 编辑结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 删除结果 |

* + - * 1. TUserSettingsController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSettingsController |
| 类功能 | | 用户鲜花期望表;(TUserSettings)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询 |
| 输入 | Page<TUser> page,  TUserSettings tUserSettings |
| 输出 | 查询结果 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 主键查询 |
| 输入 | Integer id |
| 输出 | 查询结果 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | TUserSettings tUserSettings |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 编辑数据 |
| 输入 | TUserSettings tUserSettings |
| 输出 | 编辑结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 删除结果 |

* + - 1. Service层
         1. ChatService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | ChatService |
| 类功能 | | 会话服务接口 |
| 方法1 | 名称 | public ResponseEntity<String>  sendChatCompletionRequest |
| 功能 | 发送会话 |
| 输入 | String message |
| 输出 | 会话 |

* + - * 1. EmailService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | EmailService |
| 类功能 | | 邮箱服务接口 |
| 方法1 | 名称 | public String sendEmail |
| 功能 | 发送邮件 |
| 输入 | String email |
| 输出 | 验证码 |
| 方法2 | 名称 | public boolean register |
| 功能 | 注册用 |
| 输入 | Integer id |
| 输出 | 注册结果 |
| 方法3 | 名称 | public boolean checkCode |
| 功能 | 验证验证码 |
| 输入 | String code |
| 输出 | 验证结果 |

* + - * 1. GetHexService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | GetHexService |
| 类功能 | | 哈希加密服务接口 |
| 方法1 | 名称 | public String getHex |
| 功能 | Hex加密 |
| 输入 | String str |
| 输出 | 加密后文本 |

* + - * 1. LoginService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | LoginService |
| 类功能 | | 登录服务接口 |
| 方法1 | 名称 | public TUser login |
| 功能 | 登录 |
| 输入 | String email,  String password |
| 输出 | 用户实体 |

* + - * 1. RegisterService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | RegisterService |
| 类功能 | | 注册服务接口 |
| 方法1 | 名称 | public TUser register |
| 功能 | 注册 |
| 输入 | TUser user,  String code |
| 输出 | 注册结果 |

* + - * 1. TUserContactService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserContactService |
| 类功能 | | 收货联系方式表;(TUserContact)表服务接口 |
| 方法1 | 名称 | 内部实现 |
| 功能 | 内部实现 |
| 输入 | 内部实现 |
| 输出 | 内部实现 |

* + - * 1. TUserLoginBehaviorService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserLoginBehaviorService |
| 类功能 | | 用户登录行为记录表;  (TUserLoginBehavior)表服务接口 |
| 方法1 | 名称 | 内部实现 |
| 功能 | 内部实现 |
| 输入 | 内部实现 |
| 输出 | 内部实现 |

* + - * 1. TUserSearchBehaviorService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSearchBehaviorService |
| 类功能 | | 用户搜索行为记录表;  (TUserSearchBehavior)表服务接口 |
| 方法1 | 名称 | 内部实现 |
| 功能 | 内部实现 |
| 输入 | 内部实现 |
| 输出 | 内部实现 |

* + - * 1. TUserService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSearchBehaviorService |
| 类功能 | | 用户表;(TUser)表服务接口 |
| 方法1 | 名称 | 内部实现 |
| 功能 | 内部实现 |
| 输入 | 内部实现 |
| 输出 | 内部实现 |

* + - * 1. TUserSettingsService接口

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSearchBehaviorService |
| 类功能 | | 用户鲜花期望表;(TUserSettings)表服务接口 |
| 方法1 | 名称 | 内部实现 |
| 功能 | 内部实现 |
| 输入 | 内部实现 |
| 输出 | 内部实现 |

* + - 1. Dao层
         1. TUserContactDao类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserContactDao |
| 类功能 | | 收货联系方式表;(TUserContact)表数据库访问层 |
| 方法1 | 名称 | TUserContact queryById |
| 功能 | 通过ID查询单条数据 |
| 输入 | Integer id |
| 输出 | 实例对象 |
| 方法2 | 名称 | List<TUserContact> queryAllByLimit |
| 功能 | 查询指定行数据 |
| 输入 | TUserContact tUserContact，  Pageable pageable |
| 输出 | 对象列表 |
| 方法3 | 名称 | long count |
| 功能 | 统计总行数 |
| 输入 | TUserContact tUserContact |
| 输出 | 总行数 |
| 方法4 | 名称 | int insert |
| 功能 | 新增数据 |
| 输入 | TUserContact tUserContact |
| 输出 | 影响行数 |
| 方法5 | 名称 | int insertBatch |
| 功能 | 批量新增数据 |
| 输入 | List<TUser> entities |
| 输出 | 影响行数 |
| 方法6 | 名称 | int insertOrUpdateBatch |
| 功能 | 批量新增或按主键更新数据 |
| 输入 | List<TUser> entities |
| 输出 | 影响行数 |
| 方法7 | 名称 | int update |
| 功能 | 修改数据 |
| 输入 | TUserContact tUserContact |
| 输出 | 影响行数 |
| 方法8 | 名称 | int deleteById |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 影响行数 |

* + - * 1. TUserDao类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserDao |
| 类功能 | | 用户表;(TUser)表数据库访问层 |
| 方法1 | 名称 | int insertBatch |
| 功能 | 批量新增数据 |
| 输入 | List<TUser> entities |
| 输出 | 影响行数 |
| 方法2 | 名称 | int insertOrUpdateBatch |
| 功能 | 批量新增或按主键更新数据 |
| 输入 | List<TUser> entities |
| 输出 | 影响行数 |
| 方法3 | 名称 | TUser selectByEmail |
| 功能 | 通过email查找数据 |
| 输入 | String email |
| 输出 | 实例对象 |

* + - * 1. TUserLoginBehaviorDao类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserLoginBehaviorDao |
| 类功能 | | 用户登录行为记录表;  (TUserLoginBehavior)表数据库访问层 |
| 方法1 | 名称 | TUserContact queryById |
| 功能 | 通过ID查询单条数据 |
| 输入 | Integer id |
| 输出 | 实例对象 |
| 方法2 | 名称 | List<TUserContact> queryAllByLimit |
| 功能 | 查询指定行数据 |
| 输入 | TUserLoginBehavior，  Pageable pageable |
| 输出 | 对象列表 |
| 方法3 | 名称 | long count |
| 功能 | 统计总行数 |
| 输入 | TUserLoginBehavior |
| 输出 | 总行数 |
| 方法4 | 名称 | int insert |
| 功能 | 新增数据 |
| 输入 | TUserLoginBehavior |
| 输出 | 影响行数 |
| 方法5 | 名称 | int insertBatch |
| 功能 | 批量新增数据 |
| 输入 | List< TUserLoginBehavior > entities |
| 输出 | 影响行数 |
| 方法6 | 名称 | int insertOrUpdateBatch |
| 功能 | 批量新增或按主键更新数据 |
| 输入 | List< TUserLoginBehavior > entities |
| 输出 | 影响行数 |
| 方法7 | 名称 | int update |
| 功能 | 修改数据 |
| 输入 | TUserLoginBehavior |
| 输出 | 影响行数 |
| 方法8 | 名称 | int deleteById |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 影响行数 |

* + - * 1. TUserSearchBehaviorDao类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserLoginBehaviorDao |
| 类功能 | | 用户搜索行为记录表;  (TUserSearchBehavior)表数据库访问层 |
| 方法1 | 名称 | TUserContact queryById |
| 功能 | 通过ID查询单条数据 |
| 输入 | Integer id |
| 输出 | 实例对象 |
| 方法2 | 名称 | List< TUserSearchBehavior > queryAllByLimit |
| 功能 | 查询指定行数据 |
| 输入 | TUserSearchBehavior，  Pageable pageable |
| 输出 | 对象列表 |
| 方法3 | 名称 | long count |
| 功能 | 统计总行数 |
| 输入 | TUserSearchBehavior |
| 输出 | 总行数 |
| 方法4 | 名称 | int insert |
| 功能 | 新增数据 |
| 输入 | TUserSearchBehavior |
| 输出 | 影响行数 |
| 方法5 | 名称 | int insertBatch |
| 功能 | 批量新增数据 |
| 输入 | List< TUserSearchBehavior > entities |
| 输出 | 影响行数 |
| 方法6 | 名称 | int insertOrUpdateBatch |
| 功能 | 批量新增或按主键更新数据 |
| 输入 | List< TUserSearchBehavior > entities |
| 输出 | 影响行数 |
| 方法7 | 名称 | int update |
| 功能 | 修改数据 |
| 输入 | TUserSearchBehavior |
| 输出 | 影响行数 |
| 方法8 | 名称 | int deleteById |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 影响行数 |

* + - * 1. TUserSettingsDao类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSettingsDao |
| 类功能 | | 用户鲜花期望表;  (TUserSettings)表数据库访问层 |
| 方法1 | 名称 | TUserSettings queryById |
| 功能 | 通过ID查询单条数据 |
| 输入 | Integer id |
| 输出 | 实例对象 |
| 方法2 | 名称 | List< TUserSettings > queryAllByLimit |
| 功能 | 查询指定行数据 |
| 输入 | TUserSettings，  Pageable pageable |
| 输出 | 对象列表 |
| 方法3 | 名称 | long count |
| 功能 | 统计总行数 |
| 输入 | TUserSettings |
| 输出 | 总行数 |
| 方法4 | 名称 | int insert |
| 功能 | 新增数据 |
| 输入 | TUserSettings |
| 输出 | 影响行数 |
| 方法5 | 名称 | int insertBatch |
| 功能 | 批量新增数据 |
| 输入 | List< TUserSettings > entities |
| 输出 | 影响行数 |
| 方法6 | 名称 | int insertOrUpdateBatch |
| 功能 | 批量新增或按主键更新数据 |
| 输入 | List< TUserSettings > entities |
| 输出 | 影响行数 |
| 方法7 | 名称 | int update |
| 功能 | 修改数据 |
| 输入 | TUserSettings |
| 输出 | 影响行数 |
| 方法8 | 名称 | int deleteById |
| 功能 | 删除数据 |
| 输入 | Integer id |
| 输出 | 影响行数 |

* + - 1. Entity层
         1. Tuser类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUser |
| 类功能 | | 用户表;(TUser)表实体类 |
| 属性 | private Integer uId | 用户id |
| private String uName; | 用户名 |
| private String uStatus; | 身份标识;管理员0/普通买家1 |
| private String uPassword | 用户密码 |
| private String uMail; | 用户邮箱 |
| private Object uExpect; | 购买期望 |
| private Date uCreatTime; | 创建时间 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - * 1. TUserContact类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserContact |
| 类功能 | | 收货联系方式  表;(TUserContact)实体类 |
| 属性 | private Integer id | 表id |
| private Integer uId | 用户id |
| private String ucName; | 收货人 |
| private String ucAddress; | 收货地址 |
| private String ucTel | 收货人联系方式 |
| private String uc; | 默认地址 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - * 1. TUserLoginBehavior类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserLoginBehavior |
| 类功能 | | 用户登录行为记录表;  表实体类 |
| 属性 | private Integer id | 表id |
| private Integer uId | 用户id |
| private String ulbType; | 登录or登出 |
| private String ulbTime; | 时间 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - * 1. TUserSearchBehavior类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSearchBehavior |
| 类功能 | | 用户搜索行为记录表;  (TUserSearchBehavior)表实体类 |
| 属性 | private Integer id | 表id |
| private Integer uId | 用户id |
| private String usbContent; | 搜索内容 |
| private String usbTime; | 时间 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - * 1. TUserSettings类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TUserSettings |
| 类功能 | | 用户鲜花期望表;  (TUserSettings)表实体类 |
| 属性 | private Integer id | 表id |
| private Integer uId | 用户id |
| private Integer fId; | 鲜花id |
| private Object usExpect; | 该品种鲜花期望 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - 1. Interceptor
         1. LoginConfig类

|  |  |  |
| --- | --- | --- |
| 类名称 | | LoginConfig |
| 类功能 | | 拦截器 |
| 方法1 | 名称 | public void addInterceptors |
| 功能 | 拦截器 |
| 输入 | InterceptorRegistry registry |
| 输出 | 无 |

* + - * 1. UserLoginInterceptor类

|  |  |  |
| --- | --- | --- |
| 类名称 | | UserLoginInterceptor |
| 类功能 | | 拦截器 |
| 方法1 | 名称 | public boolean preHandle |
| 功能 | 拦截器 |
| 输入 | HttpServletRequest request,  HttpServletResponse response,  Object handler |
| 输出 | 无 |

* 1. 商品模块
     1. 系统流程
        1. 公告模块设计说明

功能描述：发布公告，商家可以定义公告标题，编辑公告内容，最后发布公告；删除公告，商家可以对不需要的公告进行删除；更新公告，商家对公告的内容进行修改，包括标题和内容；查看公告，商家发布公告之后，用户和商家都可以查看公告的内容。公告模块需要在用户登录之后才能对其进行操作。

设计的数据库表：公告模块用到的数据库t\_public\_notice表。

程序运行逻辑流程图



图3-2-1-1公告模块程序运行流程图

* + - 1. 分类模块设计说明

功能描述：添加分类，商家可以添加花卉分类模块；删除分类，商家对没使用的分类进行删除处理；更新分类，商家在添加花卉分类之后，可以对分类模块进行更新处理，包括内容和标题，该操作需要在商家添加花卉分类模块之后才能进行操作，分类模块需要在用户登录之后才能对其进行操作。

设计的数据库表：分类模块用到的数据库t\_flower\_classify表。

程序运行逻辑流程图



图3-2-1-3分类模块程序运行流程图

* + - 1. 商品模块设计说明

功能描述：添加商品，商家可以对新引进的花卉上架；添加分类，商家可以把不同的花卉添加到指定的分类里面；删除商品，商家在添加花卉之后，对不需要的花卉进行下架处理，该操作需要在商家添加花卉之后才能进行操作，修改分类，对商品的分类进行修改，该操作需要在花卉已经上架并且添加分类之后才能进行操作，商品模块需要在用户登录之后才能对其进行操作。

设计的数据库表：分类模块用到的数据库t\_flower\_classify表。

程序运行逻辑流程图



图3-2-1-3商品模块程序运行流程图

* + 1. 系统管理模块类和接口设计
       1. TFlowerClassifyController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TFlowerClassifyController |
| 类功能 | | 鲜花分类(TFlowerClassify)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询所有数据 |
| 输入 | page 分页对象 tFlowerClassify 查询实体 |
| 输出 | 所有数据 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 通过主键查询单条数据 |
| 输入 | id 主键 |
| 输出 | 单条数据 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | tFlowerClassify 实体对象 |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 修改数据 |
| 输入 | tFlowerClassify 实体对象 |
| 输出 | 修改结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | idList 主键结合 |
| 输出 | 删除结果 |

* + - 1. TFlowerController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TFlowerController |
| 类功能 | | 鲜花;(TFlower)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询所有数据 |
| 输入 | page 分页对象 tFlower查询实体 |
| 输出 | 所有数据 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 通过主键查询单条数据 |
| 输入 | id 主键 |
| 输出 | 单条数据 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | tFlower实体对象 |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 修改数据 |
| 输入 | tFlower实体对象 |
| 输出 | 修改结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | idList 主键结合 |
| 输出 | 删除结果 |
| 方法6 | 名称 | public R selectByUId |
| 功能 | 根据uId查询数据 |
| 输入 | uId 用户id |
| 输出 | 多条数据 |

* + - 1. TPublicNoticeController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TPublicNoticeController |
| 类功能 | | 公告(TPublicNotice)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询所有数据 |
| 输入 | page 分页对象 tPublicNotice查询实体 |
| 输出 | 所有数据 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 通过主键查询单条数据 |
| 输入 | id 主键 |
| 输出 | 单条数据 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | tPublicNotice实体对象 |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 修改数据 |
| 输入 | tPublicNotice实体对象 |
| 输出 | 修改结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | idList 主键结合 |
| 输出 | 删除结果 |

* + - 1. TFlowerClassify类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TFlowerClassify |
| 类功能 | | 鲜花分类(TFlowerClassify)表数据库访问层 |
| 属性 | private Integer id | 序号id |
| private Integer uId | 用户id |
| private Integer fId | 鲜花id |
| private Integer cCount | 购买数量 |
| private String cStatus | 状态：0为鲜花下架，1为正常状态 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - 1. TFlower类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TFlower |
| 类功能 | | 鲜花;(TFlower)表数据库访问层 |
| 属性 | private String fId | 鲜花id |
| private String fColor | 鲜花颜色 |
| private String fImplication | 鲜花寓意 |
| private String fNumber | 鲜花数量 |
| private String fPrice | 鲜花价格 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* 1. 订单模块
     1. 系统流程
        1. 购物车模块设计说明

功能描述：商品添加，用户可以将选中的商品添加到购物车中；商品查看，用户可以随时查看购物车中的商品列表，包括商品名称、单价、数量等信息。商品修改，用户可以修改购物车中商品的数量，也可以从购物车中移除商品。结算功能，用户可以对购物车中的商品进行结算，生成订单。购物车状态保持，系统需要确保用户在不同页面或会话中购物车状态的持久性。

涉及的数据库表：该模块用到数据库t\_cart表。

程序运行逻辑流程图：



图3-3-1购物车模块程序运行流程图

* + - 1. 订单列表模块设计说明

功能描述：订单展示，订单列表模块应该能够展示所有已经下单的订单信息。每个订单通常包括订单号、下单时间、订单状态、订单金额等基本信息；订单状态管理，用户应该能够通过订单列表模块查看订单的当前状态，例如已支付、待支付、已发货、已完成等，此外，用户可能还能够对订单进行一些状态更新操作，如确认收货、取消订单等；订单详情查看，用户可以通过订单列表模块查看订单的详细信息，包括订单中的商品清单、收货地址、配送方式、支付方式等；订单搜索和筛选，订单列表模块可能提供搜索和筛选功能，允许用户根据订单号、下单时间、订单状态等条件快速查找特定订单；订单操作，除了查看订单信息外，用户可能还能够执行一些订单操作，如重新下单、申请退款、评价商品等。分页和排序，考虑到订单数量可能较大，订单列表模块通常支持分页和排序功能，以便用户浏览大量订单时能够更方便地进行导航和查看。订单统计与报表，订单列表模块可能提供一些统计和报表功能，用于分析订单数据，如销售额统计、订单量统计、用户下单趋势分析等，帮助商家更好地了解业务情况；权限管理，订单列表模块可能需要实现权限管理机制，确保只有授权的用户能够访问和操作订单信息，保护用户隐私和商家业务数据的安全。通过以上功能，订单列表模块能够为用户提供一个方便快捷的订单管理平台，帮助用户轻松地管理自己的购买记录，并为商家提供了有效的订单管理工具，有助于提升用户体验和业务效率。

涉及的数据库表：该模块用到数据库t\_order表。

程序运行逻辑流程图：



图3-3-2订单列表模块程序运行逻辑流程图

* + - 1. 订单详情模块设计说明

功能描述：用户和管理员可以查看订单的详细信息，包括订单号、收货人姓名、收货人手机、收货地址、生成时间、总价、订单状态、备注等，其中管理员可以修改收货地址和订单状态。

涉及的数据库表：该模块用到数据库t\_order\_details表。

程序运行逻辑流程图：



图3-3-3订单详情模块程序运行逻辑流程图

* + 1. 系统管理模块类和接口设计
       1. TCartController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TCartController |
| 类功能 | | 购物车;(TCart)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询所有数据 |
| 输入 | page 分页对象 tCart 查询实体 |
| 输出 | 所有数据 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 通过主键查询单条数据 |
| 输入 | id 主键 |
| 输出 | 单条数据 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | tCart 实体对象 |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 修改数据 |
| 输入 | tCart 实体对象 |
| 输出 | 修改结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | idList 主键结合 |
| 输出 | 删除结果 |

* + - 1. TorderController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TOrderController |
| 类功能 | | 订单表;(TOrder)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询所有数据 |
| 输入 | page 分页对象 tOrder 查询实体 |
| 输出 | 所有数据 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 通过主键查询单条数据 |
| 输入 | id 主键 |
| 输出 | 单条数据 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | tOrder 实体对象 |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 修改数据 |
| 输入 | tOrder实体对象 |
| 输出 | 修改结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | idList 主键结合 |
| 输出 | 删除结果 |
| 方法6 | 名称 | public R selectByUId |
| 功能 | 根据uId查询数据 |
| 输入 | uId 用户id |
| 输出 | 多条数据 |

* + - 1. TOrderDetailsController类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TOrderDetailsController |
| 类功能 | | 订单详情表;(TOrderDetails)表控制层 |
| 方法1 | 名称 | public R selectAll |
| 功能 | 分页查询所有数据 |
| 输入 | page 分页对象 tOrderDetails 查询实体 |
| 输出 | 所有数据 |
| 方法2 | 名称 | public R selectOne |
| 功能 | 通过主键查询单条数据 |
| 输入 | id 主键 |
| 输出 | 单条数据 |
| 方法3 | 名称 | public R insert |
| 功能 | 新增数据 |
| 输入 | tOrderDetails 实体对象 |
| 输出 | 新增结果 |
| 方法4 | 名称 | public R update |
| 功能 | 修改数据 |
| 输入 | tOrderDetails 实体对象 |
| 输出 | 修改结果 |
| 方法5 | 名称 | public R delete |
| 功能 | 删除数据 |
| 输入 | idList 主键结合 |
| 输出 | 删除结果 |

* + - 1. TCart类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TCart |
| 类功能 | | 购物车;(TCart)表实体类 |
| 属性 | private Integer id | 序号id |
| private Integer uId | 用户id |
| private Integer fId | 鲜花id |
| private Integer cCount | 购买数量 |
| private String cStatus | 状态：0为鲜花下架，1为正常状态 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - 1. TOrder类

|  |  |  |
| --- | --- | --- |
| 类名称 | | Torder |
| 类功能 | | 订单表;(TOrder)表实体类 |
| 属性 | private String oId | 订单id |
| private Integer uId | 用户id |
| private String oName | 收货人姓名 |
| private String oTel | 收货人电话 |
| private String oAddress | 收货地址 |
| private String oNotes | 备注 |
| private Date oCreateTime | 创建时间 |
| private Object oSumPrice | 总价 |
| private String oStatus | 订单状态;0为待付款，1为已付款未发货，2为已发货，3为待评价，4为结单 |
| private String oAddress | 收货地址 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |

* + - 1. TOrderDetails类

|  |  |  |
| --- | --- | --- |
| 类名称 | | TOrderDetails |
| 类功能 | | 订单详情表;(TOrderDetails)表实体类 |
| 属性 | private Integer id | 表id |
| private String oId | 订单号 |
| private Integer fId | 鲜花id |
| private Integer odCount | 鲜花数目 |
| 接口 | 所有属性对应的getXxx、setXxx方法 | |