|  |
| --- |
| S2023-B105 |
| 品读二手书交易平台 |
| 后台概要设计说明书 |

|  |
| --- |
| 乔轲 夏润成 王晗頔 李精文 王子骏  2023-5-9 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **《“品读”二手书交易平台后端概要设计说明书》** | | | | | | | |
| **编号** | | S2023-QK105 | | **版本** | | 2.0 | |
| **编制** | 乔轲 夏润成 王子骏 李精文 王晗頔 | | **审核** | 乔轲 | **批准** | | 乔轲 |
| **日期** | 2023.2.26 | | **日期** | 2023.2.27 | **批准日期** | | 2023.2.27 |

**后端概要设计说明书**

1. **目的**

本概要设计说明书意在为读者便于后期后端页面UI设计和页面之间跳转关系做概要说明，实现二手书资源的交易买卖，建立数据库，展示并维护销售情况。是后期“详细设计说明书”的编写依据。

预期的读者为：审核人员、小组成员及其指导老师。

1. **项目背景**

项目名称：读品-C2B2C图书交易平台。  
项目代号：S2023-QK105

项目来源：小组成员根据 “多抓鱼”图书平台的发展为灵感，共同商讨得出。

客户名称：读书爱好者

1. **参考文献**

需求说明书、立项任务书、软件开发计划书、后台概要设计

1. **总体设计**
   1. **需求规定：**
2. **硬件支持**

硬件环境：MacBook Air Apple M1 8GB

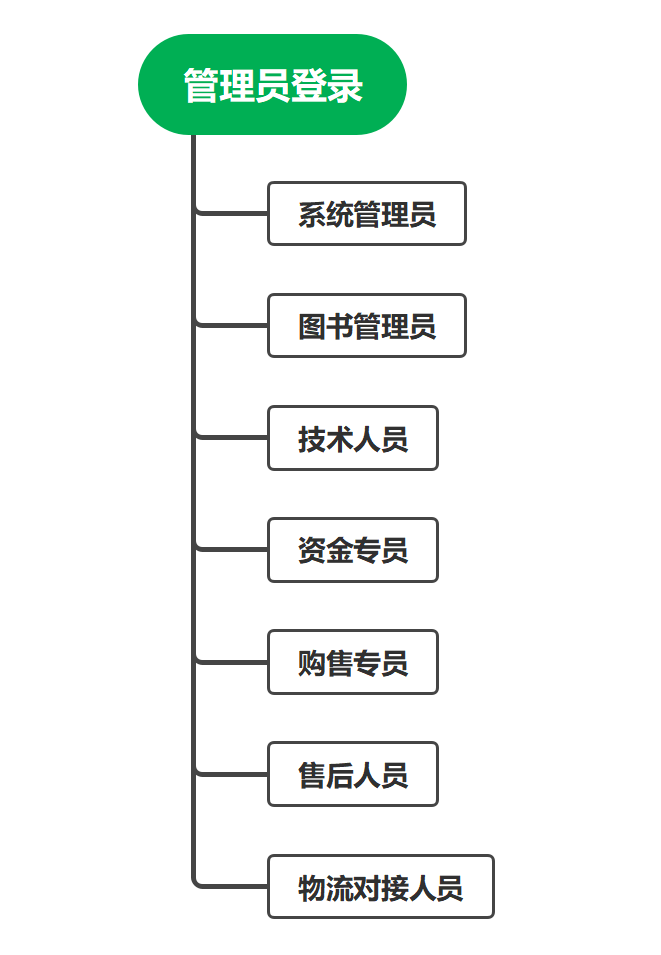
MacBook Pro (13-inch, 2017) 3.1 GHz 双核Intel Core i5 8 GB

小新13pro AMD Ryzen 7 4800U with Radeon Graphics 1.80 GHz

* 1. **运行环境：**

Windows 10，Windows11，MacOS

* 1. **项目设计概念和处理流程**



图示

描述已自动生成

**4.4结构**

业务涉及四类管理人员，分为四种管理子模块。管理员子模块暂时未明确划分具体结构，见详细设计文档。

**4.5功能需求和程序的关系**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **系统管理员** | **图书管理员** | **资金专员** | **购售专员** | **售后人员** |
| **书目信息查看** | True | True |  |  |  |
| **书目信息维护** | True | True |  |  |  |
| **用户订单查看** | True |  |  | True |  |
| **书籍折旧定义** | True |  |  | True |  |
| **图书品相判定** | True |  |  | True |  |
| **订单状态调整** | True |  |  | True |  |
| **订单咨询** | True |  | True |  | True |

5 接口设计

5.1公共接口

（1）用户的登录与注册

（2）管理员的登录

5.2外部接口：

pymysql：Django链接mysql的接口

ISBN接口：接口地址：https://api.gugudata.com/text/isbn

豆瓣接口：获取豆瓣图书信息

5.3内部接口（Web接口）：

urlpatterns = [

path('admin/', admin.site.urls),

# people

path('api/userlist/', people.views.UserList.as\_view()),

path('api/userregister/', people.views.UserRegister.as\_view()),

path('api/userlogin/', people.views.UserLogin.as\_view()),

path('api/useraddress/', people.views.UserAddressListCreate.as\_view()),

# 下面的url：pk=uid并发送get，返回此人的所有记录；pk=useraddressid并发送patch,delete，修改、删除该条记录

path('api/useraddress/<int:pk>/', people.views.UserAddressRetrieveUpdateDestroy.as\_view()),

path('api/userdetail/<int:pk>/', people.views.UserDetail.as\_view()),

path('api/userchangepassword/', people.views.UserChangePw().as\_view()),

path('api/userverifypw/', people.views.UserPwVerify.as\_view()),

path('api/adminlist/', people.views.AdminListCreate.as\_view()),

path('api/adminlogin/', people.views.AdminLogin.as\_view()),

# book

path('api/book/', book.views.BookListCreate.as\_view()),

path('api/book/<int:isbn>/', book.views.BookRetrieveUpdateDestroy.as\_view()),

path('api/bookisflow/', book.views.BookIsFlowList.as\_view()),

# shop的精髓之处

path('api/bookcatepresspricesearch/<str:pk>/<str:qk>/<str:ok>/<str:rk>/<str:sk>/',

book.views.BookRetrieveByCatePressPriceSearch.as\_view()),

path('api/bookcate/<str:pk>/<str:qk>/<str:ok>/<str:rk>/<str:sk>/',

book.views.BookCategoryListByCatePressPriceSearch.as\_view()),

path('api/bookpress/<str:pk>/<str:qk>/<str:ok>/<str:rk>/<str:sk>/',

book.views.BookPresshouseListByCatePressPriceSearch.as\_view()),

# conditionrate

path('api/conditionrate/', book.views.ConditionRateListCreate.as\_view()),

path('api/conditionrate/<int:pk>/', book.views.ConditionRateRetreiveUpdateDelete.as\_view()),

# inventory

path('api/inventory/', book.views.InventoryListCreate.as\_view()),

path('api/inventory/<int:pk>/', book.views.InventoryRetrieveUpdateDestroy.as\_view()),

path('api/inventorydetail/<int:pk>/', book.views.InventoryDetailRetrieve.as\_view()),

# cart

path('api/cart/', buy.views.CartListCreate.as\_view()),

# get方法时pk为uid，patch、delete方法时pk为自增id

path('api/cart/<int:pk>/', buy.views.CartRetrieveUpdateDestory.as\_view()),

# 购物车减少一件

path('api/cartminus/', buy.views.CartMinus.as\_view()),

# 获得一个用户的一个inventory的数量

path('api/cartinventorynum/<int:pk>/<int:qk>/<str:ok>/', buy.views.CartInventoryNumRetrieve.as\_view()),

# sell

path('api/warehouse/', sell.views.WarehouseListCreate.as\_view()),

path('api/warehouse/<int:pk>/', sell.views.WarehouseRetrieveDestroy.as\_view()),

path('api/warehouse/uid=<int:pk>/', sell.views.WarehouseOfUserRetrieve.as\_view()),

# buy\_order

path('api/buyorder/', buy.views.BuyOrderListCreate.as\_view()),

path('api/buyorder/<int:pk>/', buy.views.BuyOrderRetrieveUpdateDelete.as\_view()),

# buyorderdetail

path('api/buyorderdetail/', buy.views.BuyOrderDetailListCreate.as\_view()),

path('api/buyorderdetail/<int:pk>/', buy.views.BuyOrderDetailRetrieveUpdateDestory.as\_view()),

# sellorder

path('api/sellorder/', sell.views.SellOrderListCreate.as\_view()),

path('api/sellorder/<int:pk>/', sell.views.SellOrderRetrieveUpdateDestroy.as\_view()),

path('api/sellorder/status=<int:pk>/', sell.views.SellOrderStatusList.as\_view()),

# sellorderdetail

path('api/sellorderdetail/', sell.views.SellOrderDetailListCreate.as\_view()),

path('api/sellorderdetail/<int:pk>/', sell.views.SellOrderDetailRetrieveUpdateDestroy.as\_view()),

path('api/sellorderdetailpatch/<int:pk>/', sell.views.SellOrderDetailPatch.as\_view()),

# feedback 这两个是for user

path('api/feedback/', feedback.views.FeedbackListCreate.as\_view()),

# # GET方法时pk=uid，返回此用户的所有历史问题，包括问题和回复及处理状态

path('api/feedback/<int:pk>/', feedback.views.FeedbackRetrieveUpdateDestroy.as\_view()),

path('api/feedbackpending/', feedback.views.FeedbackListPending.as\_view()),

path('api/feedbackpendingfromasset/', feedback.views.FeedbackListPendingFromAsset.as\_view()),

path('api/feedbackfromaftersale/', feedback.views.FeedbackListFromAftersale.as\_view()),

path('api/feedbackpendingfromaftersale/', feedback.views.FeedbackListPendingFromAftersale.as\_view())

1. **系统数据结构设计**

图示, 示意图

描述已自动生成

**5.0 概述**

本项目数据存放在腾讯云MySQL数据库表中，本系统用到的数据库1个，数据表14个，数据库名称为DP。

（注：下述数据表中字段带\*且标黄的为主键，字段带$且标绿的为外键。）

**1.people**

**1.1 user（用户）**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*u\_id | \*用户码 | int | 非空 |
| u\_name | 用户名 | Varchar(100) | 非空 |
| u\_pwd | 密码(SHA256加密) | Varchar(256) | 非空 |
| u\_ma | 余额 | float | 非空 |
| u\_birth | 生日 | Varchar(10) |  |
| u\_gender | 性别 | Varchar(2) |  |

**1.2 useraddress（用户地址）**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*u\_id | \*用户码 | int | 非空 |
| receiver\_name | 用户名 | Varchar(100) | 非空 |
| receiver\_phone | 收货人电话 | Varchar(20) | 非空 |
| receiver\_region | 收货人区域 | Varchar(10) | 非空 |
| receiver\_place | 收货人地址 | Varchar(100) | 非空 |

**1.3 administrator（管理员）**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*a\_id | \*管理员id | int | 非空 |
| a\_name | 管理员名 | varchar(20) | 非空 |
| a\_pwd | 管理员密码 | varchar(20) | 非空 |
| a\_level | 管理员权限(0为超级管理员；1为图书管理员；2为资金专员；3为售后人员；4为购售专员；5为物流对接人员) | int | 非空 |
| a\_gender | 性别 | varchar(2) |  |
| a\_birth | 生日 | varchar(10) |  |
| a\_content | 简介 | varchar(1000) |  |
| a\_phone | 手机 | int |  |

1. **feedback（售后反馈表）**

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 解释 | 类型 | 非空否 |
| fba2u\_id | 表记录的主键 | int | 非空 |
| $u\_id | 用户id | int | 非空 |
| $as\_id | 售后人员id | int | 非空 |
| $as\_id | 售后人员id | int | 非空 |
| $a\_id | 资金人员id | int | 非空 |
| fba2u\_content\_u2as | 用户问题 | varchar(300) | 非空 |
| fba2u\_content\_as2u | 售后给用户反馈内容 | varchar(300) | 非空 |
| fba2u\_content\_as2a | 售后给资金专员的问题 | varchar(300) | 非空 |
| fba2u\_content\_a2as | 资金人员给售后人员的反馈 | varchar(300) | 非空 |

1. **book**

**3.1 book**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*b\_isbn | ISBN | Varchar(100) | 非空 |
| b\_name | 书名 | Varchar(100) | 非空 |
| b\_writer | 作者 | Varchar(20) | 非空 |
| b\_intro | 简介 | Varchar(1000) |  |
| b\_category | 类别 | Varchar(20) | 非空 |
| b\_presshouse | 出版社 | Varchar(50) | 非空 |
| b\_pressdate | 出版时间 | Varchar(7) |  |
| b\_price | 官方价格 | float | 非空 |
| b\_flow | 流通状态 | enum(0,1) | 非空 |

**3.2 inventory(库存)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*i\_id | 库存图书id | int | 非空 |
| $b\_ISBN | ISBN | Varchar(100) | 非空 |
| $cr\_condition | 品相 | Varchar(20) | 非空 |
| I\_book\_num | 数量 | Int | 非空 |

**3.3 mane\_regulation (图书定价规则)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*cr\_id | 自增主键 | int | 非空 |
| cr\_condition | 品相 | Varchar(20) | 非空 |
| cr\_rate | 买书折旧率 | Float | 非空 |
| S\_rate | 卖书折旧率 | float | 非空 |

1. **buy(购物相关)**

**4.1 cart(购物车)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*c\_id | \*购物车码 | int | 非空 |
| $uid | $用户id | int | 非空 |
| c\_book\_num | 图书数量 | int | 非空 |
| $i\_id | 库存图书id | Int | 非空 |

**4.2 buy\_order(购书订单)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*bo\_id | 买书订单号 | int | 非空 |
| $uid | $用户id | int | 非空 |
| Bo\_status | 订单状态(0-待处理，1-已发货，2-未审核,3已删除,4已完成) | int | 非空 |
| Bo\_submit\_time | 订单提交时间 | Datetime | 非空 |
| Bo\_finish\_time | 订单完成时间 | Datetime | 非空 |

**4.3 buy\_order\_detail**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*bod\_id | 主键 | int | 非空 |
| $bo\_id | ISBN | Varchar(100) | 非空 |
| $I\_id | 库存图书id | int | 非空 |
| Bod\_book\_num | 购书数量 | Int | 非空 |
| Bod\_order\_book\_price | 价格 | Float | 非空 |

1. **sell（卖书相关）**

**5.1 warehouse(卖书库)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*w\_id | \*卖书码 | int | 非空 |
| $u\_id | $用户id | int | 非空 |
| $B\_isbn | ISBN | Varchar(20) | 非空 |
| w\_customer\_condition | 用户选的图书品相 | Varchar(20) | 非空 |

**5.2.sell\_order(卖书订单)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*so\_id | 卖书订单号 | int | 非空 |
| $u\_id | $用户id | int | 非空 |
| So\_status | 订单状态(0-待处理，1-已发货，2-已收货,3已删除) | int | 非空 |

**5.3 sell\_order\_detail**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **解释** | **类型** | **非空否** |
| \*sod\_id | 主键 | int | 非空 |
| $so\_id | ISBN | Varchar(100) | 非空 |
| $b\_isbn | isbn | Varchar(20) | 非空 |
| W\_customer\_condition | 用户定的品相 | Varchar(20) | 非空 |
| Sod\_final\_condition | 购售定的品相 | Varchar(20) | 非空 |
| Sod\_predict\_prize | 预期价格 | Float | 非空 |
| Sod\_final\_prize | 最终价格 | Float | 非空 |

预期价格：标价x比率(s\_rate)

**6. 界面与功能设计**

1. 登陆界面Login.index.vue

图形用户界面, 应用程序

描述已自动生成

1. 售后所有问题列表Aftersaleman.allRecords.vue

**图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成**

1. 售后待处理问题列表Aftersaleman.pendingRecords.vue

图形用户界面, 应用程序

描述已自动生成

图形用户界面, 文本, 应用程序

描述已自动生成

1. 售后待反馈问题列表Aftersaleman.pendingFromAssetmanRecords.vue

**图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成**

1. **资金专员所有问题列表**

**assetman.allRecords.vue**

**图形用户界面, 文本, 应用程序

描述已自动生成图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成**

**6、 资金专员待处理问题列表**

**assetman.pendingRecords.vue**

**图形用户界面, 文本, 应用程序

描述已自动生成图形用户界面, 文本, 应用程序

描述已自动生成**

1. **购售专员买书订单列表buysaleman.buyorderinfo.index.vue**

**图形用户界面, 表格

描述已自动生成图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成**

1. **购售专员卖书订单列表buysaleman.saleorderinfo.index.vue**

**图形用户界面, 应用程序, 表格

描述已自动生成图形用户界面, 应用程序

描述已自动生成**

1. **购售专员发货buysaleman.send.vue**

**图形用户界面, 应用程序, 表格, Teams

描述已自动生成**

1. **购售专员修改价格比例buysaleman.rules.index.vue**

**图形用户界面, 应用程序

描述已自动生成**

1. **购售专员全部订单信息buysaleman.allOrders.vue**

**图形用户界面, 表格

描述已自动生成**

1. **购售专员未审核订单信息buysaleman.checkedOrders.vue**

**图形用户界面, 应用程序, 表格

描述已自动生成**

1. **购售专员审核订单buysaleman.checkWork.vue**

**图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成**

1. **购售专员已审核订单buysaleman.uncheckOrders.vue**

**图形用户界面, 表格

描述已自动生成**

1. **图书专员全部图书信息表bookman.tablemodel.vue**

**图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成图形用户界面, 应用程序

描述已自动生成图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成**