|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
|  |  |  |  |  |
| Online shop系统设计  online book shop 网上书店 |
|  |
| **REPORT FOR ASSGNMENT 6**  **2019/12/29**  71117410 旷菁宇  71117409 李鑫  71117412 冯书逸 | | | | |

|  |  |
| --- | --- |
| ***1、系统设计*** | |
| **1.1、项目范围** | 网上书店的功能主要分为匿名用户区域和会员用户功能，会员用户在匿名用户功能基础上增加了一些新功能：   1. 匿名用户：  * 欢迎界面和页面导航； * 登录账户； * 注册账户； * 找回密码； * 搜索书籍； * 查看书籍详细信息； * 购物车功能； * 寻求帮助； * 联系我们； * 查看常见问题和解答； * 网站使用策略；  1. 会员用户：  * 更改密码； * 退出账户； * 提交订单； * 查看订单记录； |
| **1.2、项目成员** | 71117410 旷菁宇  71117409 李鑫  71117412 冯书逸 |

|  |  |
| --- | --- |
| **1.3、项目环境** | * 操作系统：win10/Ubuntu 18.04； * 编程环境：visual studio code； * 网页：html+js+css； * 架构：MVC * 数据库：SQL Lite3 |
| **1.4、项目过程安**  **排** | |  |  |  |  | | --- | --- | --- | --- | | 开发过程 | 时间安排 | 目标规划 | 任务分配 | | 需求分析 | 2019/12/20-2019/12/21 | 分析整个软件的需求，完成功能需求文档 | 软件需求分析分：李鑫，旷 | | 设计阶段 | 2019/12/22-2019/12/25 | 实现数据库设计，系统设计，网页设计 |  | | 编码阶段 | 2019/12/25-2019/12/28 | 对照需求文档，编码实现完整功能的网页 |  | | 测试阶段 | 2019/12/28-2019/12/29 | 完成单元测试，集成测试，系统测试，保证系统没有明显缺陷 |  | | 部署阶段 | 2019/12/29 | 将系统部署到服务器上，可以通过域名访问网站 |  | |
| **1.5、系统架构设计** | 在需求分析中，我们已经确定了系统的功能模块，包括登录功能，注册功能，浏览功能，会员功 能和管理员功能。用户可以通过浏览器访问网站，通过使用服务器上的类库和数据库，实现图书的搜 索和购买，系统应该具有易用、美观的图形界面。  1.5.1、系统总体架构图： |
| ***2、数据库设计*** | |
| **2.1、数据表** | |  | | --- | | ticketitems:账目明细 | | ticketid | | bookinfoid：书籍信息id | | count:计数 |   表视图：   |  | | --- | | User | | id | | name:用户民 | | password：密码 |  |  | | --- | | tickets：账目 | | id | | userid:用户id | | datetime：时间 | | usermessage:用户备注 | | creditcard：卡账号 | | address:地址 | | totalprice：总价格 | | status：状态 |  |  | | --- | | bookinfo | | id | | isbn:国际书标准编号 | | title：书名 | | author:作者 | | description | | price:价格 | |
| **2.2、表及其字段的说明** | 2.2.1、用户信息表：user   |  |  |  | | --- | --- | --- | | 字段名 | 类型 | 说明 | | id | integer( primary key) | 独一无二的用户标识 | | name | text | 用户名不可重复 | | password | text | 密码不可为空 |   2.2.2、书籍信息表：bookinfo   |  |  |  | | --- | --- | --- | | 字段名 | 类型 | 说明 | | id | integer primary key | 书籍独一无二的id,主键 | | isbn | integer | 国际书标准号，不可为空，独一无二 | | title | text | 书籍名字，不可为空 | | Author | text | 作者，不可为空 | | description | text | 书籍描述 | | Price | raal | 书价格，不可为空 |   2.2.3、订单表：tickets   |  |  |  | | --- | --- | --- | | 字段名 | 类型 | 说明 | | id | integer primary key | 订单独一无二的id,主键 | | userid | integer | 用户id，外键引用自用户id | | datatime | datetime | 订单建立的时间日期，不可为空 | | usermessage | text | 用户备注信息 | | creditcard | text | 用户用于支付的银行卡账号 | | address | text | 用户收货地址 | | totalprice | real | 该订单总价格，不可为空 | | status | integer | 订单状态，未收货，未付款，已确认收货等 |   2.2.4、订单条目：ticketitems   |  |  |  | | --- | --- | --- | | 字段名 | 类型 | 说明 | | id | integer primary key | 订单独一无二的id,主键 | | ticketed | integer | 订单id，外键引用于订单表的id属性 | | bookinfoid | integer | 书籍id，外键引用于书籍信息表的id属性 | | count | integer | 计数，记录当前条目在账单中的序号，不为空 |   属性都是原子的，有唯一主键，不存在对非主键的依赖，没有传递依赖，因而是符合 3NF 的。 |
| **2.3、备份计划** | 采用手工备份计划，每天复制一份数据库文件即可 |
| **2.4、所有查询** | 2.4.1、插入  插入书籍：  INSERT INTO BOOKINFOS(id,isbINSERT INTO BOOKINFOS(id, isbn, title, author, description, price) VALUESn, title, author, description, price) VALUES(@id,@isbn,@title,@author,@description,@price)  插入用户：  INSERT INTO users(id, name, password)  VALUES (1, 'admin', 'admin234');  插入订单：  INSERT INTO tickets(id, userid, usermessage, totalprice)  VALUES (1, 1, "Init book store", 0);  插入订单项：  INSERT INTO ticketitems(id, ticketid, bookinfoid, **count**) VALUES  （@id, @ticketid, @bookinfoid, @count）  2.4.2、查询：  搜索书籍：   select isbn, title, author, description, price from bookinfos  where isbn like @q  or title like @q  or author like @q  or description like @q collate nocase  模糊搜索所有可能匹配的书籍；  搜索订单：     select \* from tickets      where userid = (          select id from users          where name = ?)  order by id desc  通过用户名得到用户id，然后搜索订单；  搜索订单条目          select b.title, b.isbn, i.count from ticketitems as i          inner join bookinfos as b on b.id = i.bookinfoid          where i.ticketid = ?          order by i.id asc  通过订单条目序号展示该条目，包括书的名字，isbn编号，序号； |
| ***3、软件设计与实现*** | |
| **3.1、软件架构** | 网站采用了经典的MVC，三层BS架构，M为Model，V为View，，C为controller   * Model(模型)，-模型代表一个存取数据的对象，它可以带有逻辑，在数据变化时实时更新； * View(视图) -视图代表模型包含的数据可视化； * Controller(控制器) -控制器作用于模型和视图上。他控制数据流向模型对象，并在数据变化时实时更新视图，它使视图与模型分开； |
| **3.2、模块说明** | Model：    -schema：用来建数据库  -seed：确定对表进行初始化  -database：连接数据库  View：    -account：账户视图  signin：登录页面  signup：注册页面  -checkout：支付视图  -help：帮助视图  contact：联系我们  faq：常见疑问以及解答  policy：使用策略  -home：主页视图  cart：购物车视图  detail：详细视图  home：导航到主页视图  search：搜索视图  -share：一些共享页面视图  error：发生错误的视图  forbidden：没有权限访问文件时返回视图  macros：宏  navbar：导航条视图  notfound：404视图  userstate：显示用户信息的视图  -layout：页面布局，一个综合上述html文件的容器  Controller：  -account：负责管理账户信息，新建帐户，查看用户名，登陆账户等；  -cart：购物车的管理，包括增删购物车，查看购物车；  -checkout：订单支付；  -detail：查询书籍详细信息；  -home：展示主页面；  -orders：查询订单信息；  router：导航模块；  search：搜索数籍模块； |

~~Main contents~~

~~Database design~~

Function design

~~Software design and implement~~

~~Database query~~