



网上书店系统 《系统设计》

学 院： 软件学院

学 生 姓 名： 71119101 杨状凌

71119102 王博杰

71119103 许 润

71119104 包亦成



SOUTHEAST UNIVERSITY

第一部分 项目概述

1、服务器

- (1) OS: Windows Server 2019
- (2) 500GB 以上的硬盘空间，不包括数据存储空间
- (3) JDK: jdk9
- (4) DB: MySql
- (5) 开发平台: IntelliJ Idea、Visual Studio Code

2、客户机器环境

- (1) Intel、AMD K5 或以上微处理器 (CPU)；
- (2) Windows、Mac、Linux 等操作系统；
- (3) 内存不少于 4G、硬盘空间不小于 80G；
- (4) WEB: 要求 IE8 以上版本，最好 IE10 以上版本。

3、开发人员

71119101 杨状凌、71119102 王博杰
71119103 许 润、71119104 包亦成

4、项目计划

过程	时间	目标	任务分配
需求阶段	2021/12/9	完成需求分析	杨状凌、许润
设计阶段	2021/12/10-12/11	完成系统设计	包亦成、王博杰
实现阶段	2021/12/12-12/23	完成页面与编码	前端：包亦成、杨状凌 后端：王博杰、许润
测试阶段	2021/12/24	测试系统	全员参与
完善阶段	2021/12/25-12/26	完善系统	全员参与

第二部分 设计约束

一、需求约束

1、本系统应当遵循的技术标准

本项目遵循 HTML5、CSS3 标准，JavaScript 采用 ES6 语法标准，服务端采用 SpringBoot 框架标准。

2、软、硬件环境标准

本系统采用 B/S 架构。使用 HTML, JavaScript, CSS, Java 编写，数据库采用 MySQL。系统部署在云服务器中。

3、接口/协议标准

本系统通过 HTTP 协议实现客户端与服务器之间的数据通信。

4、用户界面标准

使用 Vue 组件库 ElementUI，并进行界面全部美工优化。

5、软件质量

(1) 正确性

系统必须交易能够被正确处理；

(2) 健壮性

系统应能够 7*24 小时无故障运行；

(3) 效率性

系统可以支持 100 个终端同时发起业务，处理业务的时间不超过 10 秒钟；

(4) 易用性

界面应采用图形化操作方式，便于业务人员操作；

(5) 安全性

报文中的关键数据域以密文的方式传输；

(6) 可扩展性

应该充分考虑到将来交易的修改或增加，避免需求变更时大规模修改程序。

二、隐含约束

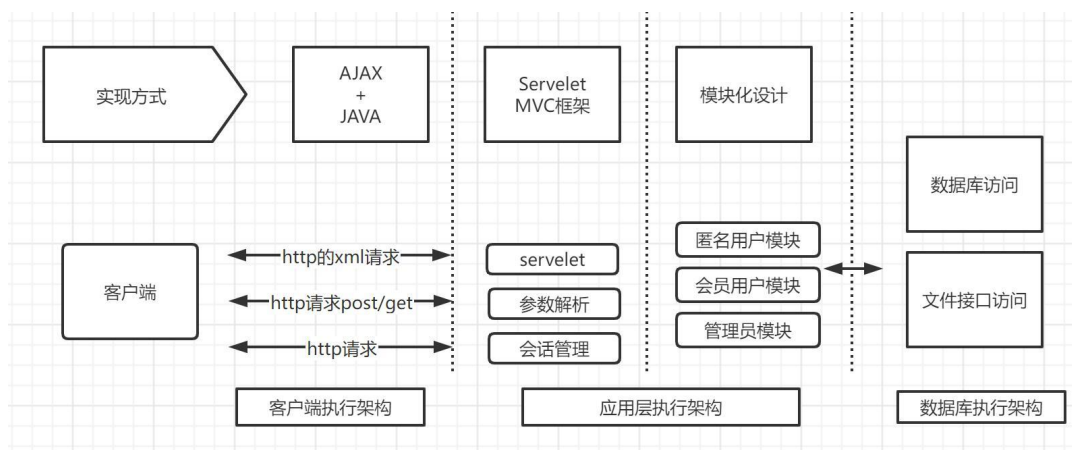
(1) 用户具有基本的业务技能和基本的电脑知识，对我们提供的操作界面应保证他们经过简单培训后无障碍的操作；

(2) 软件可以流畅运行在 IOS4.0 版本以上的设备中；

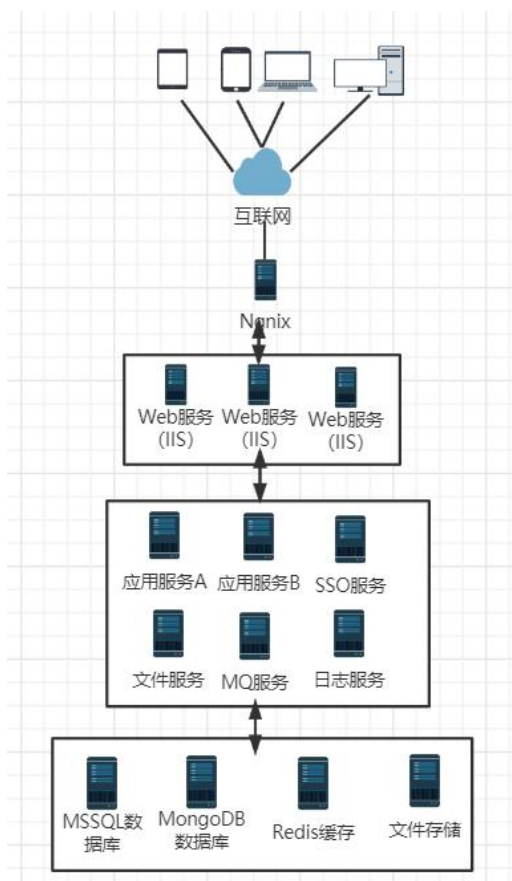
(3) 应该把有可能变动的参数存放到配置文件或数据库中，保证修改参数的灵活性。

第三部分 总体设计

一、逻辑架构设计

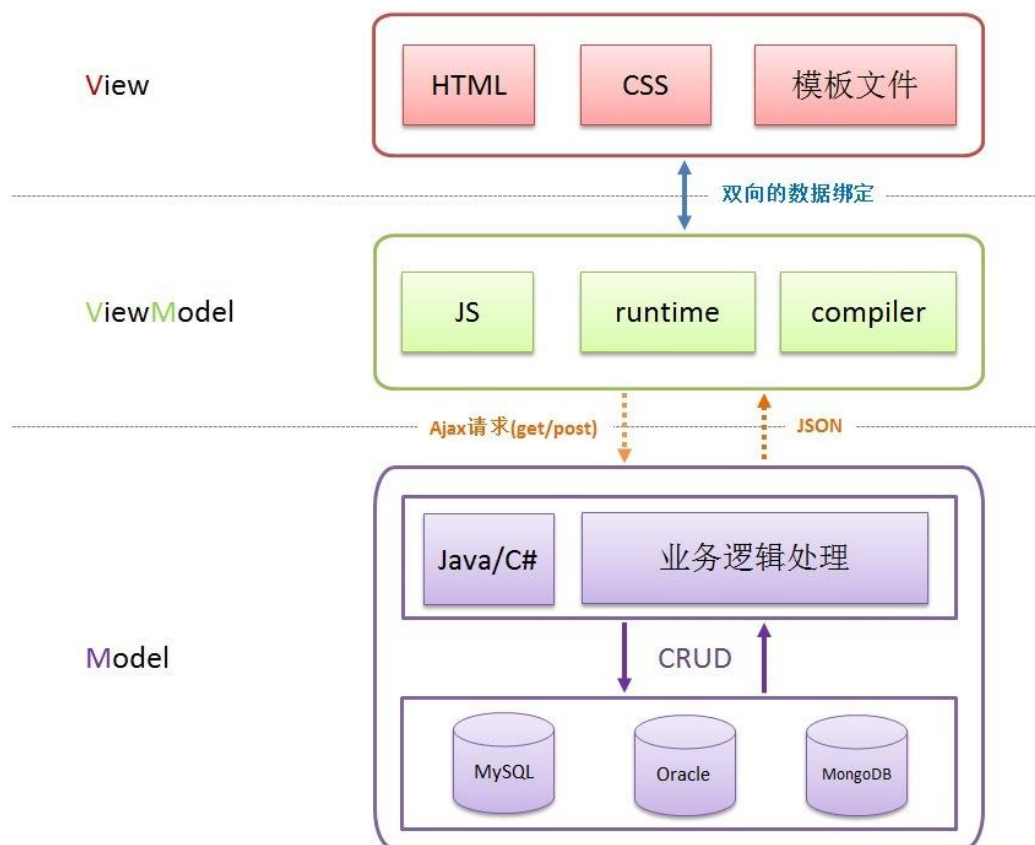


二、物理架构设计



三、技术架构设计

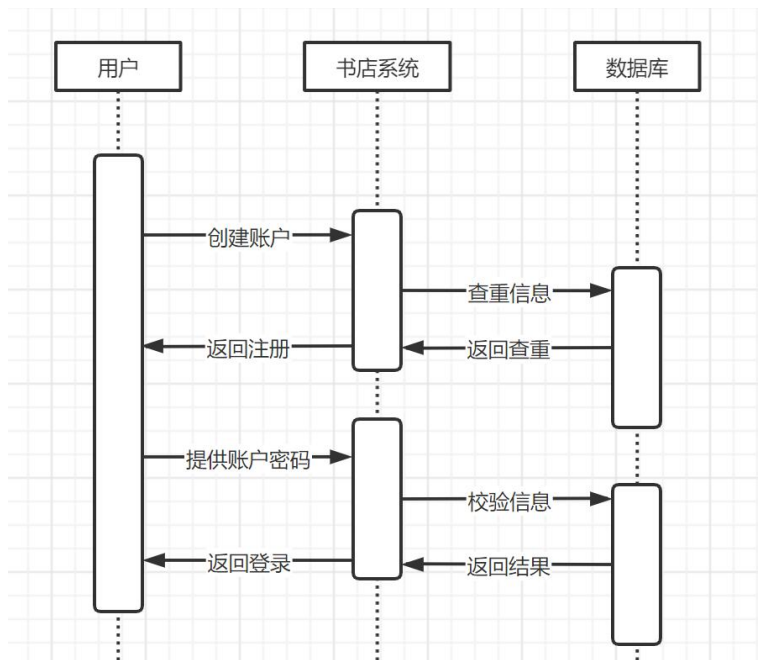
本系统采用 MVVM 架构，这是一种新型的软件架构模式，其本质上是 MVC 的改进版。MVVM 有助于将图形用户界面的开发与后端业务逻辑的开发分离开来，MVVM 的视图模型是一个值转换器，这意味着视图模型负责从模型中转换数据对象，以便轻松管理和呈现对象，视图模型实现了中间者的功能。



第四部分 方案设计

一、系统相关业务流程

1、注册、登录流程（匿名用户）



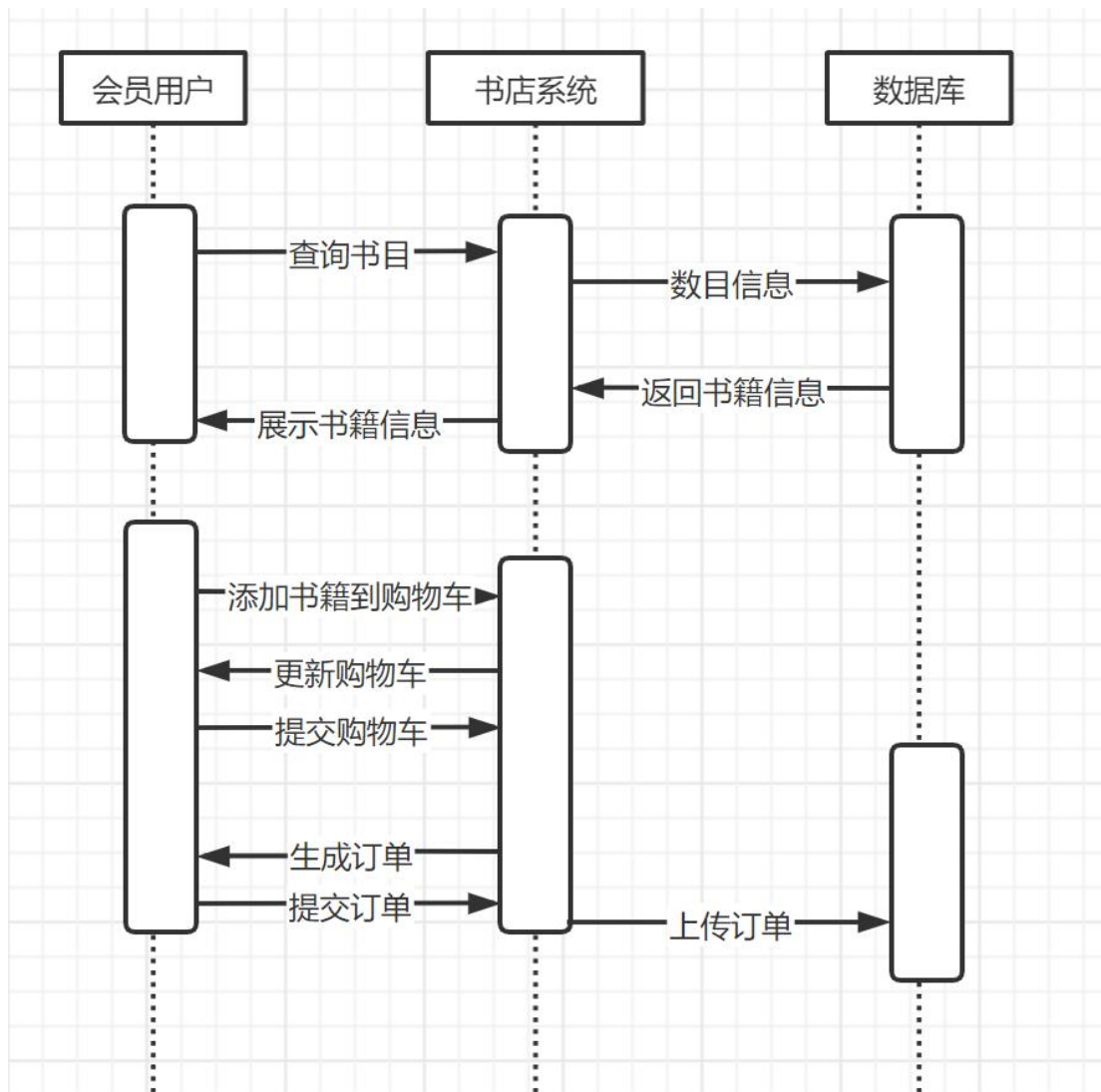
注册说明：

1. 用户首先发起创建账户请求。
2. 系统中接受创建账户的请求，并向数据库申请获得查重信息。
3. 数据库返回查重信息给书店系统，验证成功后向用户返回注册信息。

登录说明：

1. 用户首先发起登录请求，并提供账户、密码。
2. 系统中接受登录的请求，并向数据库申请获得校验信息。
3. 数据库返回校验结果给书店系统，验证成功后向用户返回登录信息。

2、查询、购物车、订单流程（会员用户、匿名用户）



查询说明：

1. 用户首先发起查询书目请求。
2. 系统中接受查询请求，并向数据库中获取书目信息。
3. 数据库返回书籍信息给书店系统，书店系统向会员用户展示书籍信息。

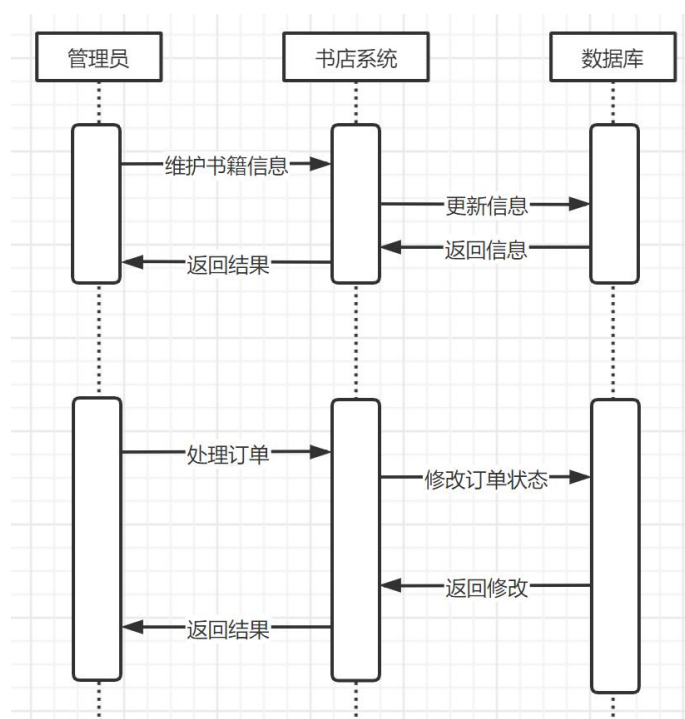
添加购物车说明：

1. 用户首先发起添加书籍至购物车的请求。
2. 系统中接受添加的请求，并更新购物车。

订单说明：

1. 用户首先发起提交购物车的请求。
2. 系统中接受提交的请求，并生成订单，返回至用户界面。
3. 用户确定订单后，订单提交至书店系统，系统将订单上传至数据库。

3、维护书籍、处理订单流程（系统管理员）



维护书籍说明：

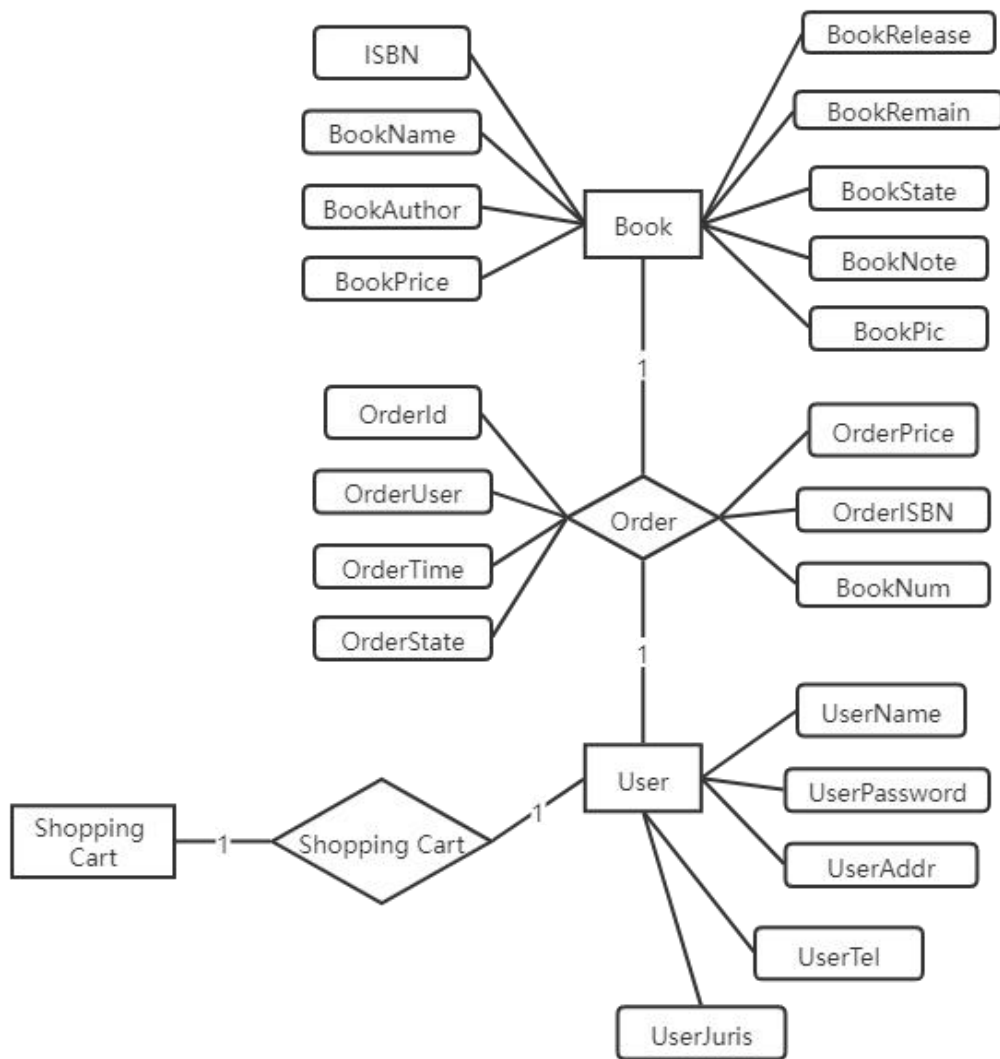
1. 管理员首先发起维护书籍（增、删、改、查）的请求。
2. 书店系统接受维护请求，并将更新信息提交到数据库。
3. 数据库更新完成后返回更新信息至系统。
4. 系统返回结果至管理员界面。

处理订单说明：

1. 管理员首先发起处理订单的请求。
2. 书店系统接受维护请求，并将修改订单信息提交到数据库。
3. 数据库更新完成后返回修改信息至书店系统。
4. 系统返回结果至管理员界面。

第五部分 E-R 实体设计

一、E-R 图



二、实体描述

1、Book 实体描述

编号	英文名	中文名	数据类型
1	ISBN	书籍编号	String
2	BookName	书籍名称	String
3	BookAuthor	书籍作者	String
4	BookPrice	书籍价格	Float
5	BookRemain	剩余库存	Int
6	BookRelease	出版时间	String
7	BookState	书籍状态	Boolean
8	BookNote	书籍简介	String
9	BookPic	书籍图片	String

2、User 实体描述

编号	英文名	中文名	数据类型
1	UserName	用户名	String
2	UserPassword	用户密码	String
3	UserAddr	用户地址	String
4	UserTel	用户电话	String
5	UserJuris	用户权限	Boolean

3、Order 实体描述

编号	英文名	中文名	数据类型
1	OrderId	订单编号	Int
2	OrderUser	下单用户	String
3	OrderTime	下单时间	String
4	OrderState	订单状态	Boolean
5	OrderPrice	订单价格	float
6	OrderISBN	书籍编号	String[]
7	BookNum	书籍数量	Int[]

第六部分 数据库设计

对于数据库表单的设计,为了更好地体现订单与书籍的一对多关系,我们在实体类的基础上,增加了 orderBook 表单。

1、Book 表单

编号	英文名	数据类型	备注
1	ISBN	nchar (13)	主键 不可空
2	BookName	varchar (255)	不可空
3	BookAuthor	varchar (255)	不可空
4	BookPrice	Float	不可空
5	BookRemain	Int	不可空
6	BookRelease	dateTime	不可空
7	BookState	tinyint	状态包括 0: 正在销售、1: 暂时下架 不可空
8	BookNote	text	不可空
9	BookPic	varchar (255)	不可空

2、User 表单

编号	英文名	数据类型	备注
1	UserName	varchar (255)	主键 不可空
2	UserPassword	varchar (255)	不可空
3	UserAddr	varchar (255)	不可空
4	UserTel	nchar (11)	不可空
5	UserJuris	tinyint	表示用户权限: 0: 会员用户、1: 管理员 不可空

3、Order 表单

编号	英文名	数据类型	备注
1	OrderId	Int	主键 不可空
2	OrderUser	varchar (255)	不可空
3	OrderTime	dateTime	不可空
4	OrderState	tinyint	状态有：0 未处理、1 已接受、2 已拒绝 不可空
5	OrderPrice	float	不可空

4、OrderBook 表单

编号	英文名	数据类型	备注
1	OrderId	Int	主键 不可空
2	ISBN	nchar (13)	主键 不可空
3	BookNum	Int	不可空