



软件体系结构设计(SAD)

——网上书店管理系统的软件体系结构设计

组名：光宗耀组

组员：刘菲，杨玥莹，王雨晨，蒋安琪，任雨晴

指导老师：余仲星

目录

1 引言.....	4
1.1 标识.....	4
1.2 系统概述.....	4
1.3 文档概述.....	5
1.4 基线.....	6
2 引用文件.....	7
3 CSCI 级设计决策.....	7
3.1 设计决策.....	7
3.2 质量保证决策.....	8
3.3 工程管理决策.....	9
4 CSCI 体系结构设计.....	9
4.1 体系结构.....	9
4.1.1 程序(模块)划分	9
4.1.2 程序(模块)层次结构关系	10
4.1.3 架构描述.....	11
4.1.4 软件单元.....	11
4.1.5 架构观点.....	12
4.2 全局数据结构说明.....	13
4.2.1 常量.....	13
4.2.2 变量.....	14
4.2.3 数据结构.....	15
4.3 CSCI 部件.....	17
4.4 执行概念.....	19
4.5 接口设计.....	20
4.5.1 接口标识和接口图.....	21
4.5.2 登陆界面的接口图.....	21
4.5.3 员工界面的接口图.....	22

4.5.4 管理员界面的接口图	23
5 CSCI 详细设计	23
5.1 体系结构框架和体系结构描述语言	23
5.2 详细设计	23
6 需求的可追踪性	25
7 注解	26
7.1 背景信息	26
7.2 术语定义	27
附录	28
附录 A.图表	28
A.1 程序划分设计图	28
A.2 登陆界面的接口图	28
A.3 员工界面的接口图	30
A.4 管理员界面的接口图	30

1 引言

1.1 标识

标识号：OBMS-2404

软件名称：网上书店管理系统

缩略词语：Online bookstore management system

版本号：v1.1.1

发行号：20240414

1.2 系统概述

本项目的是建立一个功能全面的书店管理系统，着眼于为书店业务提供全方位的支持和优化。在书本信息管理模块中，除了基本的录入、修改、查询、统计和打印功能外，还考虑实现批量导入、自动化更新书籍信息以及与供应商系统的集成，以确保书籍信息的及时性和准确性。在消费者信息管理模块中，除了常规的买书、查看书本信息、扣款等操作外，还可以引入会员管理功能，包括会员注册、积分管理和会员特权服务，从而提升客户忠诚度和满意度。而数据分析模块除了支持销售情况的查询和统计外，还可以实现销售预测、库存管理优化等功能，以帮助书店管理人员做出更明智的决策。

在系统功能外，还考虑实现跨平台支持，例如提供 Web 端、移动端和桌面端的访问方式，以满足不同用户群体的需求。同时，加强系统的可扩展性和灵活性，允许根据不同书店的特点和需求进行定制化配置和扩展功能。此外，关注用户体验和界面设计，使系统操作简单直观，提升用户的使用体验和满意度。

在安全性方面，除了提供数据备份、恢复和权限管理等基本功能外，还需要加强对用户数据的保护和隐私保密，采取有效的加密和安全认证措施，确保用户信息的安全性。同时，建立健全的审计日志系统，及时发现和应对潜在的安全威胁和异常行为。

总之，本项目旨在打造一个功能可靠、用户友好的书店管理系统，为书店业务的数字化转型提供有力支持，提升书店的竞争力和服务水平。

1.3 文档概述

软件体系结构设计（SAD）是软件工程中至关重要的一环，它提供了对软件系统设计的高层次视图和框架。这一设计包括系统架构、模块设计、接口设计和数据结构设计等多个方面。SAD 的编写基于对需求分析和系统设计的深入理解，是对系统整体结构和各个部分之间关系的详细描述。

在软件体系结构设计中，系统架构是其中最为核心的部分之一。它描述了系统的整体结构，包括各个模块之间的层次关系、通信方式以及数据流动。系统架构的设计直接影响到系统的性能、可扩展性和可维护性。

除了系统架构，模块设计也是 SAD 中的重要内容之一。模块设计将系统划分为各个功能模块，每个模块负责实现系统中特定的功能或服务。这种模块化的设计使得系统更易于开发、测试和维护。

另外，接口设计也是 SAD 的关键组成部分。接口定义了不同模块之间的通信规范和数据传递方式。良好设计的接口能够提高系统的灵活性和可重用性，同时降低模块之间的耦合度。

最后，数据结构设计也是 SAD 中不可或缺的一环。数据结构设计定义了系统中所使用的各种数据类型和数据结构，包括数据库表结构、数据存储方式等。合理的数据结构设计能够提高系统的数据处理效率和数据访问速度。

综上所述，软件体系结构设计是软件开发过程中至关重要的一环，软件体系结构通常包含以下内容：

1. 系统架构设计：描述系统的整体架构、模块划分、模块功能、模块之间的关系和数据流等。
2. 模块设计：对系统设计的各个模块进行设计和描述，包括每个模块的功能、接口、输入输出、数据结构、算法和程序流程等。
3. 接口设计：描述模块之间的接口规范，包括接口类型，系统接口设计以及通信的协议、接口参数、接口传递的数据类型和格式等。
4. 数据结构设计：描述系统所使用的各种数据结构，包括数据结构的类型、结构定义、数据元素的类型和结构、数据的存储和访问方式等。
5. 安全设计：描述系统的安全性设计，包括系统的保密性、完整性、可用性等，用加密算法对用户密码等敏感信息加密以及确保数据完整的数据校验机制。

6. 测试设计：描述系统的测试策略和测试计划，包括测试方法、测试环境、测试数据和测试用例等，覆盖系统的各个功能以及场景

7. 部署设计：描述系统的部署设计和实施方案，根据系统要求进行软件安装，包括数据库、应用服务器等，对系统进行配置，包括数据库配置、服务器配置等。

1.4 基线

设计基线在软件工程中具有关键意义，它是软件开发过程中的重要标志和里程碑。作为一种基础文档或者说参考线，设计基线承载着软件设计方案的核心要素，并在整个软件生命周期中发挥着指导和支持作用。

设计基线的建立基于软件体系说明 (SAD) 和软件架构设计文档，这些文档详细描述了软件系统的各个方面，包括系统需求、结构、功能、性能等。设计基线将这些信息整合并加以规范化，形成了具体的设计方案，为后续的开发、测试、维护和升级工作提供了重要依据。

设计基线不仅仅是一份静态文档，更是软件开发过程中的活跃参与者。它随着软件开发的进行不断演化和完善，反映了团队对系统需求和设计的理解以及对技术选型和架构方案的决策。在软件开发的阶段，设计基线可以被用来指导开发人员编写代码、测试人员进行测试、维护人员进行修改和升级，从而确保软件系统的一致性、可维护性和可扩展性。

总之，设计基线是软件开发过程中的重要产物，它为整个团队提供了一个统一的、可追溯的设计框架，为软件项目的成功交付奠定了坚实的基础。

系统设计如下计划：

①系统设计文档：该模块需要包括书本信息的增删改查功能，可以参考 SRS 中的书本管理需求进行模块划分和接口设计。具体的实现可以采用 MVC 架构模式，将模型层、视图层和控制层进行分离。

②数据库设计文档：该模块可以设计一个名为 “buyers” 的表来存储消费者信息。该表需要包括消费者号、消费者昵称、消费者年龄、手机号码等字段。同时，为了提高查询效率，可以在姓名字段上建立索引；同理，给书本，出版社订单等都建表。

③用户界面设计文档：该模块需要提供用户友好的订单管理界面，可以设计一个书本列表页面。书本列表页面需要展示书本的基本信息包括梗概，图片等以及当前库

存状态，并提供搜索和过滤功能。书本购买详情页面需要展示所购买书本的详细信息，同时提供编辑和删除功能。

④系统架构设计文档：该模块需要与其他模块进行协同工作，可以通过定义接口来实现模块之间的交互。同时，为了保证系统的可扩展性和可维护性，可以采用面向对象的设计原则，将系统划分为多个独立的模块，并采用设计模式来解决常见的软件设计问题。

⑤系统测试文档：该模块需要进行单元测试、集成测试和系统测试等多个阶段的测试。单元测试需要测试每个模块的基本功能和边界条件，集成测试需要测试模块之间的接口和交互，系统测试需要测试整个系统的性能、稳定性和安全性。这样，通过以上的设计基线，可以保证开发团队对于该模块的设计、实现和测试都具有清晰的目标和规范，并且能够确保该模块满足 SRS 中的需求，同时提高开发效率和软件质量。

2 引用文件

[1]杨丹,戴玉敏.基于计算机软件的数据库编程技术[J].电子技术与软件工程,2018(09):154

[2]钟睿.基于计算机软件的数据库编程技术[J].计算机产品与流通,2018(02):32.

[3][美]Shari Lawrence Pfleeger、[加]Joanne M·Atlee 软件工程(第4版·修订版)

3 CSCI 级设计决策

3.1 设计决策

设计决策：采用分层架构设计模式来组织网上书店管理系统的软件结构。

决策理由：分层架构是一种常用的设计模式，它将系统划分为多个层次，每个层次具有特定的职责和功能，以提高系统的可维护性、扩展性和重用性。

决策细节包括下面几点：

用户界面层：设计用户友好的界面，与用户进行交互并展示系统的功能和数据。通过优秀的界面元素和布局，提供良好的用户体验，使用户能够轻松地浏览书籍、下单购买等操作。

应用逻辑层：处理用户界面传递的请求，并根据业务规则进行逻辑处理。这一层包括业务逻辑、数据验证和处理用户输入等功能，确保系统在处理用户请求时能够有效地执行必要的操作。

数据访问层：负责与数据库进行交互，包括数据的读取、写入和查询等操作。采用合适的数据访问技术，如 ORM 或 SQL 查询，以提供对数据的持久性访问，并保证数据的完整性和一致性。

数据库层：用于存储和管理系统的数据，包括数据库的设计和建模、表结构定义以及数据关系的管理。通过有效地组织和管理数据，确保系统能够高效地存储和检索所需的信息。

第三方集成层：如果需要与其他系统或服务进行集成，设计一个专门的集成层，负责处理与外部系统的通信和数据交换。采用适当的协议和接口，如 API 或 Web 服务，实现系统与外部系统的无缝集成，提高系统的整体性能和扩展性

3.2 质量保证决策

决策：进行系统功能测试以确保其正确性和一致性。

决策理由：为了确保网上书店管理系统的功能符合用户需求并且工作正常，需要进行系统功能测试来验证系统的正确性和一致性。

决策细节：执行以下测试活动：单元测试：对各个模块和组件进行单独测试，以验证其功能和逻辑的正确性。

集成测试：将各个模块和组件组合起来进行测试，确保它们能够正确地协同工作。

系统测试：对整个网上书店管理系统进行综合测试，验证其功能、性能、可用性和安全性等方面的要求。

用户验收测试：由用户参与的测试活动，以确认系统是否满足其需求并能够实际应用。

异常情况测试：测试系统在异常情况下的行为和处理能力，例如输入错误、网络故障等。

性能测试：评估系统在不同负载和压力条件下的性能表现，包括响应时间、吞吐量和并发用户数等指标。

3.3 工程管理决策

决策：采用敏捷开发方法进行项目管理和团队协作。

决策理由：敏捷开发方法可以提高开发团队的灵活性、效率和响应能力，适应需求变化和快速交付可用的软件产品。

决策细节：执行以下工程管理决策措施：敏捷开发方法：采用敏捷开发方法 Scrum 进行项目管理和团队协作。这种方法强调迭代开发、持续集成和团队合作，以快速适应变化和提供高质量的软件。

用户参与：积极地与用户进行沟通和合作，以确保对用户需求的准确理解和及时反馈。通过持续的用户参与和反馈，可以及时调整项目方向和优先级，确保交付符合用户期望的产品。

迭代开发：将开发过程划分为多个迭代周期，每个周期交付一部分可用的功能。每个迭代周期都包括需求分析、设计、开发、测试和部署等活动，以增量方式构建和完善系统。

持续集成：通过自动化测试和持续集成工具，确保团队成员的代码能够快速集成和测试，减少集成问题和软件缺陷，并提高团队的协作效率。

进度管理：使用项目管理工具和技术，如甘特图、看板和迭代计划等，对项目进度和任务分配进行监控和管理，确保项目按时交付和资源合理利用。

风险管理：识别、评估和管理项目中的风险，并制定相应的风险应对策略。持续跟踪和监控项目中的风险，并及时采取措施以降低风险对项目的影响。

团队协作：通过良好的沟通、协作和团队建设活动，促进团队成员之间的合作和有效的知识共享，提高团队的效率和工作质量。

4 CSCI 体系结构设计

4.1 体系结构

4.1.1 程序(模块)划分

消费者模块：

这一部分主要完成与消费者数据库和书本数据库的交互，基于 SQL 提供一些基本操作的接口，可以完成对消费者数据库中信息的查询、修改、插入和删除等，即消

消费者可以通过注册会员增加信息条目或是修改个人信息；与书本数据库的交互则是消费者浏览图书页面选择是否购买图书，改变书本数据库。

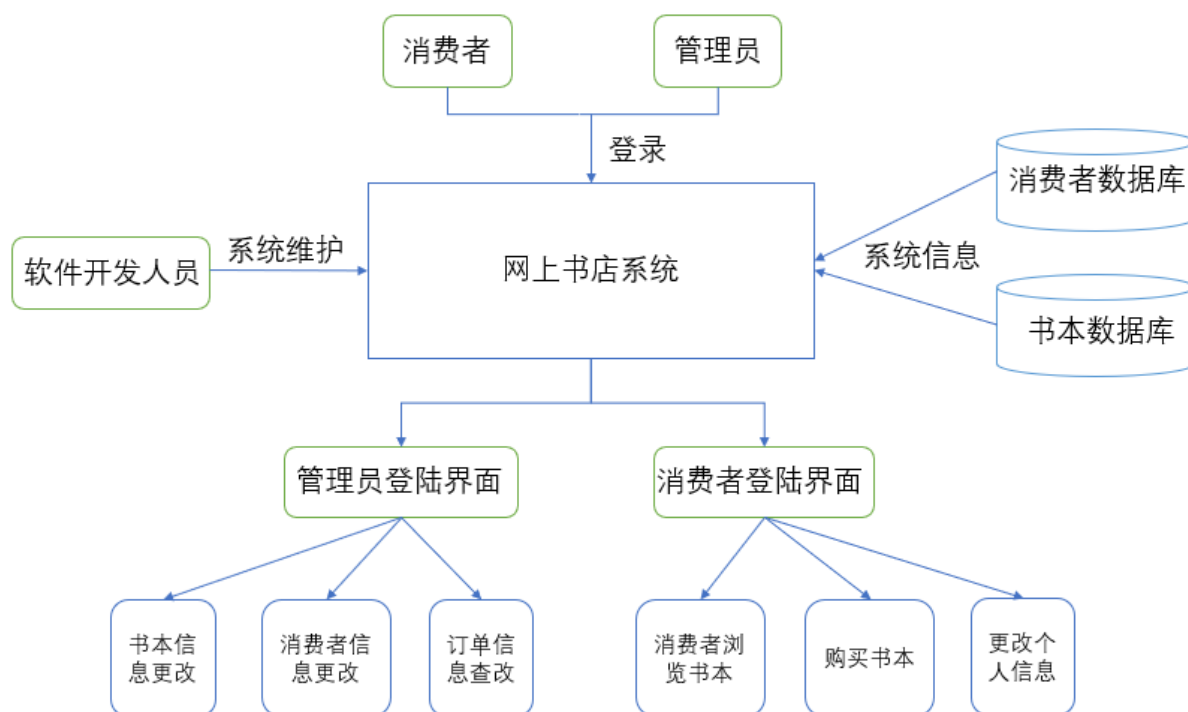
管理员模块：

这一部分主要完成与消费者数据库和书本数据库的交互。管理员可以通过消费者数据库查询注册用户的个人信息，进行用户管理。可以完成对书本数据库信息的查询、修改、插入和删除等，完成对于书本的管理。

软件开发人员维护模块：

软件开发人员可以对系统的数据库等其他模块进行维护。

程序划分设计图如下：



4.1.2 程序(模块)层次结构关系

在网络书店的系统中，各个模块之间相互交织，宛如一张密不可分的网。用户管理模块和图书管理模块之间息息相关，用户通过前者注册登录，然后才能畅游于后者的海洋中，查找并购买自己心仪的图书。而购物车管理模块与图书管理模块紧密相连，当用户将图书放入购物车时，这两者就像手牵手一样，共同完成购物的使命。订单管理模块则与购物车管理模块密切相依，一旦用户确定购物车的内容，订单管理模

块就会接手，把购物车中的图书转变为订单，同时更新数据库中的相关信息。管理员管理模块与用户管理模块和图书管理模块也是紧密相连的，管理员可以通过它来管理用户信息和图书信息，需要时与用户管理模块和图书管理模块进行数据交流和操作。数据库模块则像是这个系统的核心，负责存储和管理所有数据，各个模块都需要与它进行交互，确保信息的流通和更新。这些模块之间的紧密联系构成了系统的基础，为用户提供了流畅、高效的购书体验。

4.1.3 架构描述

用户界面层 (UI Layer)：作为系统与用户直接交互的前端界面，用户界面层实现了诸如用户注册、图书浏览、购买图书等功能。其主要责任是接收用户输入，并将请求传递至下一层进行处理。

业务逻辑层 (Business Logic Layer)：业务逻辑层承担着处理用户请求、实现系统各项业务逻辑的责任。该层涵盖了用户管理、图书管理等功能的实现，并负责与数据库层进行交互，以获取或更新数据。

数据访问层 (Data Access Layer)：数据访问层负责与数据库之间的交互，包括建立数据库连接、执行数据查询、更新等操作。该层将数据库操作封装成接口，以供业务逻辑层调用。

系统维护层 (System Maintenance Layer)：系统维护层包括系统监控、日志记录、错误处理等功能。此层的存在有助于管理员监视系统运行状态，并及时发现并解决问题。

4.1.4 软件单元

基于系统的体系结构模块和软件架构，给出具体的软件单元：

用户界面单元 (UI Unit)：用户界面单元负责呈现给用户的界面，包括注册、登录、图书浏览、购买图书等功能的界面设计和实现。用户界面应简洁直观，用户友好，以提升用户体验。

用户管理单元 (User Management Unit)：用户管理单元负责处理用户的注册、登录、个人信息管理等功能。它包括用户注册验证、用户登录验证、个人信息修改等功能的实现。

图书管理单元 (Book Management Unit): 图书管理单元负责管理系统中的图书信息, 包括图书的添加、编辑、删除、查询等功能。它还负责实现图书的分类、排序、推荐等功能, 以使用户能够方便快捷地浏览和购买图书。

购物车管理单元 (Shopping Cart Management Unit): 购物车管理单元负责管理用户的购物车信息, 包括将图书添加到购物车、从购物车删除图书、修改购物车中图书的数量等功能。它还负责计算购物车中图书的总价和总数量, 以使用户确认订单。

订单管理单元 (Order Management Unit): 描述: 订单管理单元负责处理用户的订单信息, 包括生成订单、查看订单详情、取消订单等功能。它还负责向用户提供订单状态跟踪、支付功能等服务, 确保用户能够顺利完成购买流程。

数据库访问单元 (Database Access Unit): 数据库访问单元负责与数据库进行交互, 包括连接数据库、执行 SQL 查询、更新数据库等操作。它将数据库操作封装成接口, 供其他模块调用, 以实现数据库的访问和操作。

系统维护单元 (System Maintenance Unit): 系统维护单元负责监控系统的运行情况, 记录系统日志, 处理异常和错误等。它还负责系统的配置管理、性能优化、安全管理等工作, 以保证系统的稳定性和安全性。

4.1.5 架构观点

当我们面对一个复杂的问题时, 往往需要将其分解成更小、更可管理的部分。服务导向架构 (SOA) 的核心理念就是将业务功能模块化, 以一种松散耦合的方式构建系统。这些独立存在的模块通过明确定义的接口进行交互。

实施 SOA 体系结构意味着将这些独立的服务部署在不同的计算机上, 它们相互通信以完成系统的整体任务。每个服务都是独立的, 只需关注自己的职责。这意味着可以定期更新、改进性能和修复漏洞, 而不会影响整个系统的稳定性和可用性。

SOA 体系结构的另一个重要优势是服务的松散耦合。这意味着单个服务的故障或问题不会对整个系统造成崩溃或失效。与紧密耦合的设计相比, 这种松散耦合的设计使得更新和扩展新功能变得更加容易。SOA 鼓励使用标准化接口来定义服务之间的通信方式, 例如使用基于 HTTP 的 RESTful API 或基于 SOAP 的 Web 服务。这种标准化接口有助于不同技术栈的服务之间进行无缝集成, 并降低了集成的复杂性。也倡导异步通信模式, 使得服务可以在不同时间、不同速率下进行通信, 从而提高系统的灵活性

和响应性。例如，可以使用消息队列来实现异步消息传递，降低了服务之间的耦合度。

此外，SOA 体系结构还具有可重用性，开发的服务可以在其他应用程序和系统中重复使用，而无需担心兼容性问题。这提高了效率，减少了成本和风险。

最后，SOA 体系结构简化了技术的部署和管理。因为每个服务都是独立工作的，所以更容易进行维护和控制，降低了系统的复杂性。因为 SOA 强调的是一种面向服务的架构风格，从而实现系统的高度灵活性和可扩展性。

4.2 全局数据结构说明

4.2.1 常量

1. 使用到的常量数据文件名称及其所在目录：

2. 用到的常量

书籍类别常量：

FICTION：小说类别

NON_FICTION：非小说类别

REFERENCE：参考书类别

CHILDREN：儿童书类别

MAGAZINE：杂志类别

AUDIO_BOOK：有声书类别

EBOOK：电子书类别

COMICS：漫画类别

POETRY：诗集类别

库存状态常量：

IN_STOCK：库存中

OUT_OF_STOCK：缺货

LOW_STOCK：库存紧张

BACK_ORDERED：预定

DISCONTINUED：停售

RETURNED：退货

用户角色常量：

ADMIN：管理员

CUSTOMER: 顾客

GUEST: 游客

支付状态常量:

PAYMENT_PENDING: 待支付

PAYMENT_SUCCESS: 支付成功

PAYMENT_FAILED: 支付失败

REFUNDED: 已退款

优惠类型常量:

PERCENTAGE_DISCOUNT: 百分比折扣

FIXED_AMOUNT_DISCOUNT: 固定金额折扣

BUY_ONE_GET_ONE: 买一送一

FREE_SHIPPING: 免运费

系统配置常量:

TAX_RATE: 税率

SHIPPING_COST: 运费

MAX_BOOKS_PER_ORDER: 每单最大书籍数量限制

MAX_BOOKS_IN_STOCK: 最大库存容量限制

4.2.2 变量

1. 可能涉及到的数据文件名称及其所在目录:

①会员信息数据文件:

文件名称: vip.dat

存放目录: /data/vip

②书本信息数据文件:

文件名称: book.dat

存放目录: /data/book

③权限信息数据文件:

文件名称: permission.dat

存放目录: /data/permission

④数据备份文件:

文件名称: backup.dat

存放目录: /data/backup/

4.2.3 数据结构

1. vip (员工信息)

功能说明: 用于存储会员基本信息的数据结构。

具体数据结构说明: 包含会员得编号、姓名、性别、年龄、联系方式等信息。

定义:

```
struct vip {  
    int vipId; // 员工编号  
    string name; // 姓名  
    string gender; // 性别  
    int age; // 年龄  
    string phone; // 联系方式  
};
```

2. book(书本信息)

功能说明: 用于书本的价格、编号等信息的数据结构。

具体数据结构说明: 包含书本编号、价格、出版, 作者等信息。

定义:

```
struct Salary {  
    int bookId; // 员工编号  
    float price; // 基本工资  
    string writer; //作者  
    publisher; //出版社  
    mainstory; //梗概  
};
```

3. orders (订单)

功能说明: 用于存储每个成功提交的订单的信息的数据结构。

具体数据结构说明: 包含会员编号、书本编号、书本价格、购买时间、总价、等信息。

定义:

```
struct orders{
    string vipid; //会员编号
    dic buy; //购买的书本
    datetime time; //时间
    float finalpric; // 总价
};
```

4. Backup (数据备份)

功能说明：用于存储系统数据备份的数据结构。

具体数据结构说明：包含备份日期、备份时间、备份文件名等信息。

定义：

```
struct Backup {
    string backupDate; // 备份日期
    string backupTime; // 备份时间
    string backupFileName; // 备份文件名
};
```

功能说明：用于存储系统数据备份的数据结构。具体数据结构说明：包含备份日期、备份时间、备份文件名等信息。

定义：

```
struct Backup {
    string backupDate; // 备份日期
    string backupTime; // 备份时间
    string backupFileName; // 备份文件名
};
```

5. Permission (权限管理)

功能说明：用于存储系统权限管理的数据结构。

具体数据结构说明：包含用户类型、用户权限等信息。

定义：

```
struct Permission {
    string userType; // 用户类型
```



```
        string userPermission; // 用户权限  
    };
```

4.3 CSCI 部件

1. 其 CSCI 软件配置项如下：

数据库：包括会员信息、书本信息和订单信息等。

用户界面：提供交互式的操作界面，使得用户可以方便地进行各类操作。

会员管理模块：实现会员信息变更等功能，帮助书店进行网上销售。

书本管理模块：实现书本的上架，下架，信息更改等功能，保障书店的正常工作。

订单管理模块：实现订单的更新功能。

系统管理模块：实现系统权限管理、日志记录、安全措施等功能，保证系统的安全性和稳定性。

2. 本系统的软件配置项静态关系如下：

用户界面依赖于书本管理模块和用户管理模块以及订单管理模块，因为用户需要在界面上进行的操作需要通过这三个模块来实现相应功能。

会员管理模块和书本管理模块以及订单管理模块都依赖于数据库，因为它们需要对其中存储的信息进行读取、修改、删除等操作。

系统管理模块则是独立模块，它不依赖其他配置项的功能，但包括它自己的功能和数据会被其他配置项使用和调用。

总体而言，本系统采用了模块化设计的思想，实现了各模块之间的松耦合，易于维护和扩展。

3. 以下是该书本管理系统中各软件配置项的用途：

数据库：作为存储所有会员信息、书本信息和订单信息的核心组件，提供数据访问服务，支持对数据进行查询、修改、删除等操作。

用户界面：提供可视化的交互式操作界面，让会员自己可以直观地管理自己的账户信息。它与其他系统模块交互，把用户输入输出传递给底层模块来完成相关操作。

会员管理模块：实现会员信息变更等功能，帮助书店进行网上销售。

书本管理模块：实现书本的上架，下架，信息更改等功能，保障书店的正常工作。

系统管理模块：包括系统权限管理、日志记录、安全措施等，确保系统的稳定性和安全性，防止未经授权或错误操作造成的系统故障及信息泄露等问题。同时，能够追踪管理人员及会员在系统上的操作状况，方便出现问题后的处理和调查。通过这五个软件配置项的协同配合，该书本管理系统可以完成全面、高效，安全的会员信息管理以及书店内部信息的统计分析等功能。

4. 以下是该书本管理系统中各软件配置项的开发状态/类型：

数据库：开发完成，会根据需要进行调整和优化。

用户界面：正在开发，在 UI 设计、实现及测试等方面持续改进，并且需要与其他配置项进行充分融合测试。

书本管理模块：已开发完成，并已适当测试。它是该书本管理系统的核心部分之一，但在未来可能需要进行调整以满足不断变化的需求。

会员管理模块：已开发完成，并已经进行充分的测试。它是该书店管理系统的重要组成部分之一。

系统管理模块：正在开发。

总体来说，该书店管理系统还在开发中，在未来也会进行各方面的升级和调整以应对时代变化。

5. 以下是该网上书店管理系统中各软件配置项计划使用的计算机硬件资源：

数据库：由于需要存储大量信息和相关数据，因此需要一台高性能的服务器来处理数据库的读写操作。其中 CPU、内存、磁盘空间等方面的要求可能会根据数据量的变化而有所调整。

用户界面：主要运行在计算机的图形用户界面（GUI）上。对于 GUI 运行环境的要求相对较低，通常通过一个普通的桌面或者笔记本电脑就可以满足要求。

书本管理模块：可能需要一个中央处理单元（CPU）速度较快、内存充足的计算机环境，以确保其正常运行和快速响应。

会员管理模块和订单管理模块：它需要进行大量的计算和数据读写，因此需要高性能的 CPU 以及大量的内存来完成工资计算、记录管理等操作。

系统管理模块：需要具备较高的安全可靠性和，因此对于硬件资源也有较高的要求。

例如，需要稳定的电源保障、自动备份机制等。

总体来说，在不同的组件中，根据功能、需求和实现方式的不同，其计算机硬件资源需求是有所区别的。决定哪些硬件资源需要使用以及使用的规格，需要根据具体情况进行灵活调整。

4.4 执行概念

执行控制流：在计算机软件中，CSCI（Computer Software Configuration Item）之间的执行控制流通常包括顺序执行、条件分支和循环等结构。举例来说，在书店的会员信息管理模块中，用户可以选择进行录入、修改、查询、统计或打印自己的相关信息等操作，系统会根据用户的选择执行相应的控制流程。

数据流：CSCI 之间的数据流常被用来传递信息和状态，以实现不同模块的交互。举例而言，在书店会员信息管理模块中，用户选择录入个人信息时，录入界面会接收用户输入的信息，并将其传递给后台数据处理模块。

动态控制序列：CSCI 之间的动态控制序列被用于实现复杂的交互行为，比如一些高级操作或工作流程。例如，在订单管理模块中，用户提交购买书籍的订单，订单需经过书店确认后才能发货。

状态转换图：状态转换图描述了 CSCI 的状态及其之间的转换关系，通常用于描述状态机或有限状态自动机。比如，在会员信息管理模块中，一个会员的信息可能处于不同的状态，比如已录入、已停用等，而书籍也有类似的状态，比如已上架和已下架，它们之间可能存在状态转换关系。

时序图：时序图描述了 CSCI 之间的时序关系，即各个配置项之间的交互顺序和时间。比如，在数据分析模块中，会员可以按照不同的条件查询书籍信息，查询结果将在一定时间内返回并展示给会员。

配置项之间的优先关系：配置项之间可能存在一些优先关系，即某些配置项需要在其他配置项执行之前执行。例如，在数据备份模块中，备份操作必须在数据恢复操作之前执行，以保证数据的完整性和可靠性。

中断处理：中断处理通常用于处理一些异常情况，比如系统崩溃或用户取消操作等。在这些情况下，系统可能需要执行一些特殊的处理程序，以保障系统的安全性和稳定性。

异常处理：异常处理用于应对一些不可预期的异常情况，比如输入错误，比如会员或书籍的编号与我们设计的规则不符。在这种情况下，系统需要采取相应的措施以处理异常情况，确保系统的正常运行。

4.5 接口设计

1. 用户接口：

用户交互体验应该是用户友好的，确保用户能够轻松地进行操作。通过图形用户界面（GUI），用户可以便捷地输入查询书籍信息、修改自己的会员资料以及删除账号等操作。管理员用户应该享有更多的权限，能够方便地浏览和管理会员信息以及书籍信息，并可以根据不同的条件进行排序和过滤，以快速找到所需信息。

为了提高用户操作效率，系统应该提供明确的菜单结构和快捷键，使用户可以方便地访问各个功能模块。这样，用户就能快速地导航到他们所需的功能，提升整体的使用体验。

此外，用户应该能够轻松地打印和导出书籍信息或会员信息的查询和统计结果，以便进一步分析或备份数据。这样，用户就能够更方便地利用系统生成的信息。

为了帮助用户更好地理解和使用系统，系统应该提供详尽的帮助文档和用户手册。这些文档应该清晰地解释系统的各个功能和操作步骤，并提供示例和提示，帮助用户更好地利用系统完成工作。

总的来说，用户交互体验应该是直观、高效和用户友好的。通过提供便捷的操作方式、清晰的导航结构和详尽的帮助文档，系统可以确保用户能够充分利用系统的功能，提升工作效率和满意度。

2. 硬件接口：

系统应该能够运行在 Windows 或 Linux 等常见操作系统上。

系统应该能够支持标准的输入和输出设备，如键盘、鼠标、打印机、显示器等。

系统应该能够支持数据库服务器，如 MySQL、Oracle 等。系统应该能够支持网站服务器，如 Apache、Tomcat 等。

3. 软件接口：

系统应该能够与数据库服务器进行通信，并能够使用 SQL 语言进行数据操作。

系统应该能够与其他系统进行数据交换，如 ERP 系统、HR 系统等。

系统应该能够使用标准的 API，如 RESTful API 等。

系统应该能够使用第三方组件或库，如 JQuery、Bootstrap 等。

4. 通信接口的需求：

系统可能支持 HTTP、HTTPS 协议数据传输

系统应该能够支持 SOAP 协议进行 Web 服务调用。

系统应该能够支持 SMTP 协议发送邮件。

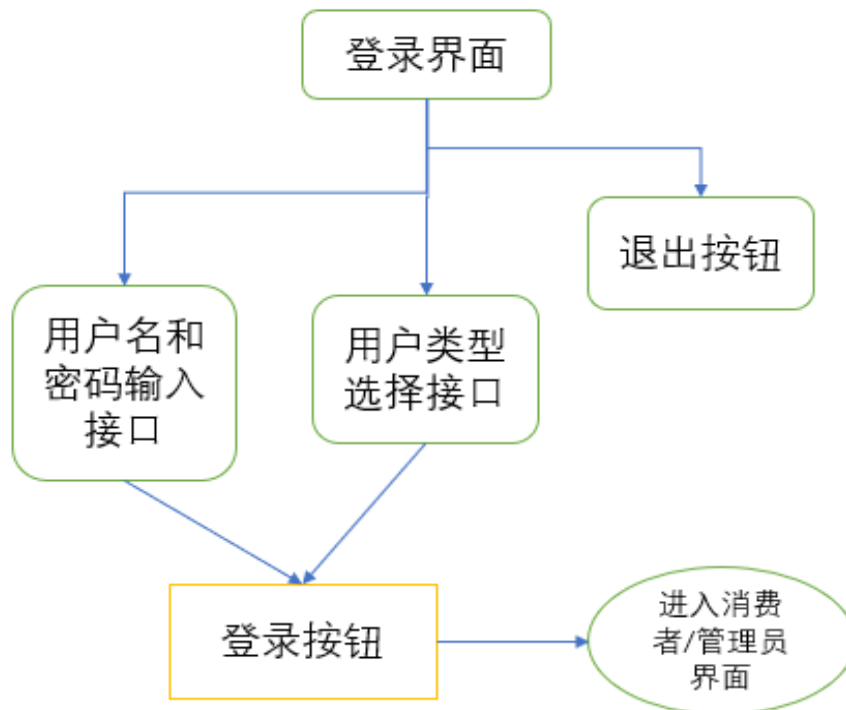
系统应该能够支持 FTP 协议进行文件传输。

4.5.1 接口标识和接口图

本条标识所需的 CSCI 外部接口，也就是 CSCI 和与它共享数据、向它提供数据或与它交换数据的实体的关系。该标识应说明哪些实体具有固定的接口特性(因而要对这些接口实体强加接口需求)，哪些实体正被开发或修改(从而接口需求已施加给它们)。可用一个或多个接口图来描述这些接口。

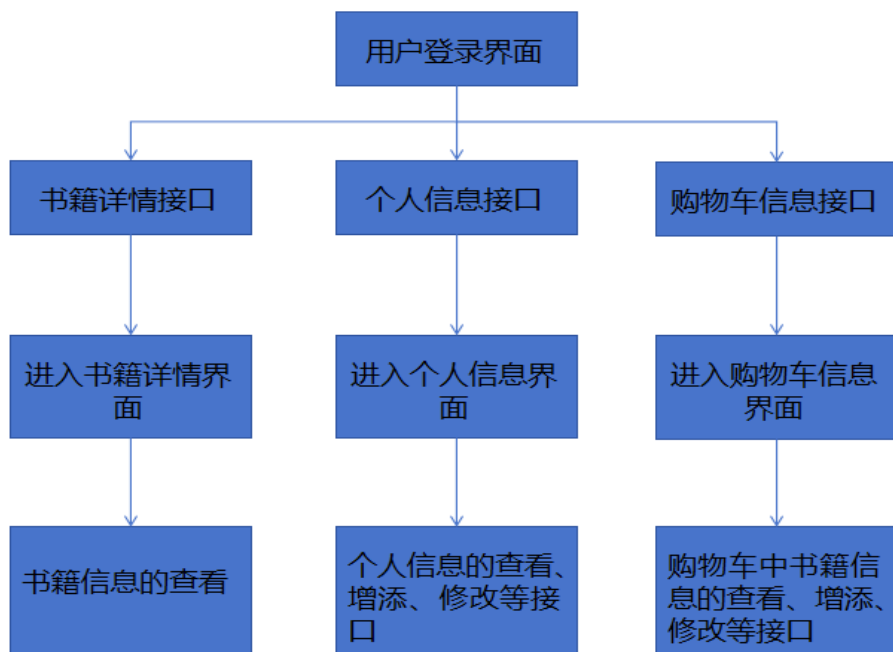
4.5.2 登陆界面的接口图

登陆界面的接口图如下：



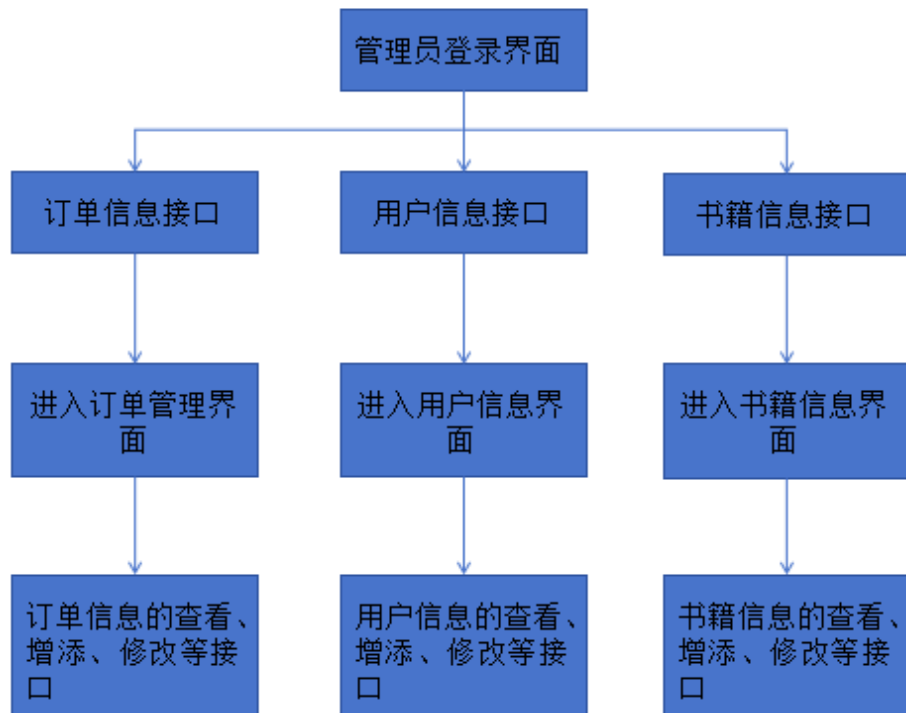
4.5.3 员工界面的接口图

员工界面的接口图如下：



4.5.4 管理员界面的接口图

管理员界面的接口图如下：



5 CSCI 详细设计

5.1 体系结构框架和体系结构描述语言

以下是该书店管理系统的体系结构框架：

Presentation Layer：用户界面，包括用户登录和操作界面。

Business Logic Layer：业务逻辑层，处理用户请求，调用数据访问层来访问数据库。

Data Access Layer：数据访问层，负责与数据库交互，包括增删改查等操作。

Database Layer：数据库层，存储所有的数据。

5.2 详细设计

a. 客户端软件配置项需要满足以下约束和限制：

客户端需要连接到服务器进行数据交互

客户端需要实现界面交互，如输入用户信息、查询会员信息、显示会员列表等客户端需要保证数据传输的安全性和完整性

b. 客户端软件配置项中包含过程式命令和一些命令行工具或者脚本来辅助管理和操作系统，如：

查询会员信息或者书本信息的 SQL 语句更新书本信息和会员信息的 SQL 语句

显示员工列表的 GUI 构造器的输入

用户权限管理命令：包括添加新的会员、修改会员信息、删除会员、会员权限管理等命令。

系统配置管理命令：包括修改系统参数、配置数据备份、数据恢复、系统维护等命令。

数据库备份工具：用于备份人事管理系统所使用的数据库，包括会员信息、书本信息，订单信息等。

数据库恢复工具：用于恢复备份的数据库文件。

数据库初始化脚本：用于在系统首次启动时初始化数据库，包括创建数据库表、添加默认数据等。

数据库升级脚本：用于在系统升级时更新数据库结构，保证旧数据的兼容性。

系统日志脚本：用于记录系统日志，包括用户登录信息、操作日志等。

e. 网上书店管理系统的 CSCI 应包含以下数据元素：

书本信息：包括书名、作者、出版信息等

会员信息：姓名、性别、出生日期、联系方式、ID 号、注册时间等

订单信息：下单时间、支付时间等

系统设置信息：包括系统参数设置、用户权限设置、系统日志等记录。

客户端软件配置项包含输入、输出和其他数据元素，如：

输入数据：会员的基本信息，如姓名、性别、出生日期、联系方式等

输出数据：会员的基本信息，如姓名、性别、出生日期、联系方式等

其他数据元素：会员的购买记录等

f. 客户端启动时需要先连接服务器客户端界面交互的响应包括：

输入会员信息查询会员信息显示会员列表响应时间包括：数据转换

数据传送操作客户端运行期间的操作序列和动态控制序列包括：

序列控制方法：事件驱动

该方法的逻辑与输入条件：客户端交互事件

a) 数据在内存中的进出：通过网络连接与服务器进行数据交互

b) 离散输入信号的感知：客户端交互事件

异常与错误处理包括：

a) 输入数据格式错误或缺失：应该对输入数据进行格式检查，确保其满足数据类型和范围的要求。如果输入数据缺失，则应该提示用户提供必要的的数据。

b) 数据库连接失败：如果连接数据库失败，则应该显示错误消息，并提示用户检查数据库设置和网络连接。

c) 数据库操作失败：如果数据库操作失败，例如插入或更新数据失败，或者查询数据不存在，则应该显示错误消息，并提示用户重新尝试或联系管理员。

d) 系统资源不足：如果系统资源不足，例如内存或磁盘空间不足，则应该显示错误消息，并提示用户释放一些资源或联系管理员。

e) 未知异常：如果发生未知异常，应该记录错误信息，并提示用户联系管理员。

6 需求的可追踪性

a. 从软件配置项到 CSCI 需求：

软件配置项：网上书店管理系统用户接口

CSCI 外部接口需求：接口必须在 Windows 和 Linux 平台下运行

CSCI 内部接口需求：用户登录接口必须与会员信息管理接口交互

CSCI 内部数据需求：需要在系统中存储和会员个人信息

适应性需求：用户界面必须易于使用，可自定义

保密性和私密性需求：会员的个人信息必须保密

CSCI 环境需求：无特别需求

计算机软件需求：需要安装 MySQL 数据库和 Qt 框架

计算机通信需求：系统需要与企业内部服务器进行通信

软件配置项：会员信息管理模块

CSCI 能力需求：模块必须能够添加、修改和删除会员信息

CSCI 内部接口需求：模块必须与用户接口交互

CSCI 内部数据需求：需要在系统中存储和管理会员个人信息

适应性需求：会员信息必须可以按不同的类别进行管理

保密性和私密性需求：会员的个人信息必须保密

软件配置项：书本信息管理模块

CSCI 能力需求：模块必须能够添加、修改和删除书本信息

CSCI 内部接口需求：模块必须与会员接口交互

CSCI 内部数据需求：需要在系统中存储和管理书本信息

适应性需求：书本信息必须可以按不同的类别进行管理

保密性和私密性需求：没有必须需要保密的内容

软件配置项：订单信息管理模块

CSCI 能力需求：模块必须能够查看订单信息，可删除

CSCI 内部接口需求：模块必须与会员接口交互

CSCI 内部数据需求：需要在系统中存储和管理订单信息

适应性需求：可查询

保密性和私密性需求：订单的唯一 ID 号需要保密

7 注解

7.1 背景信息

随着互联网的普及，越来越多的消费者倾向于在线购物。网上书店作为一种新型的购书方式，为广大读者提供了便捷、快速的购书体验。与传统实体书店相比，网上书店具有以下优势：

节省时间和精力：读者无需亲自前往书店，只需轻点鼠标即可浏览和购买图书；

价格优惠：网上书店通常提供较低的价格折扣，吸引消费者；

丰富的图书资源：网上书店的图书种类繁多，满足不同读者的需求；

个性化推荐：根据读者的购买历史和兴趣爱好，为其推荐合适的图书。

7.2 术语定义

用户：指在网上书店进行注册、登录、浏览和购书的消费者；

店主：指负责网上书店的运营和管理的人员，包括进书、售书、库存、账目和客户管理等工作；

会员：指在网上书店注册并取得唯一用户名的用户，享有购书等特权；

分级浏览：指网上书店根据图书分类，为用户提供不同级别的浏览权限；

库存管理：指店主对网上书店的图书库存进行实时监控和调整，确保图书供应充足；

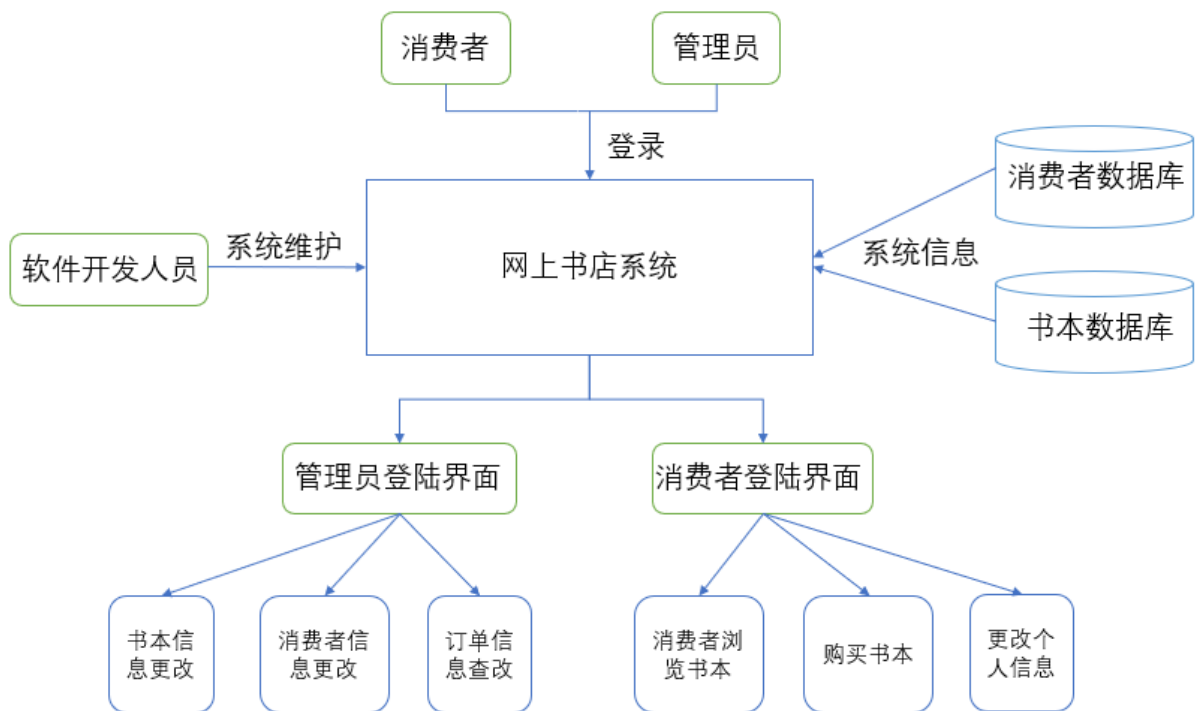
账目管理：指店主对网上书店的财务状况进行记录和分析，包括收入、支出、利润等；

客户管理：指店主对网上书店的用户进行管理，包括客户留言、客户反馈、客户满意度调查等。

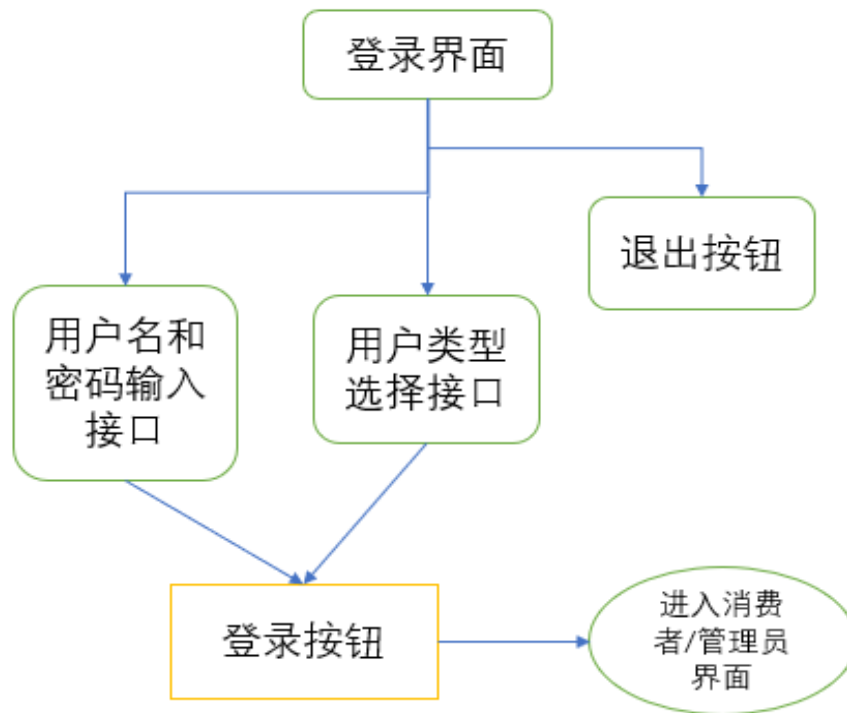
附录

附录 A.图表

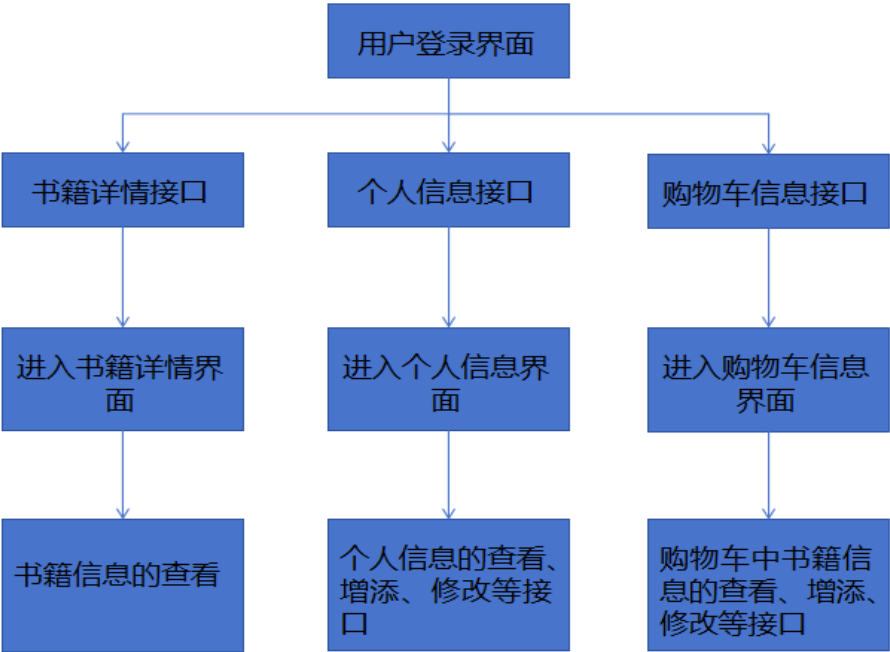
A.1 程序划分设计图



A.2 登陆界面的接口图



A.3 员工界面的接口图



A.4 管理员界面的接口图

