目 录

[一、引言 1](#_Toc1657)

[1.1编写目的 1](#_Toc27966)

[1.2背景 1](#_Toc31312)

[1.3定义 2](#_Toc25897)

[1.4参考资料 2](#_Toc11230)

[二、实际开发结果 2](#_Toc18869)

[2.1产品 2](#_Toc17214)

[2.2主要功能和性能 5](#_Toc8869)

[2.2.1功能划分 5](#_Toc4317)

[2.2.2图书分享 6](#_Toc4000)

[2.2.3图书借阅 6](#_Toc11011)

[2.2.4图书查阅 6](#_Toc30214)

[2.2.5图书搜索 6](#_Toc20708)

[2.2.6图书推荐 6](#_Toc4681)

[2.2.7图书收藏 6](#_Toc1806)

[2.2.8图书评论 7](#_Toc12565)

[2.2.9货币系统 7](#_Toc16490)

[2.2.10用户认证 7](#_Toc14042)

[2.2.11管理员 7](#_Toc16432)

[2.3进度 7](#_Toc4204)

[三、开发工作评价 8](#_Toc1141)

[3.1对生产效率的评价 8](#_Toc15825)

[3.2对产品质量的评价 8](#_Toc17989)

[3.3对技术方法的评价 8](#_Toc28805)

[3.4开发中出错原因的分析 9](#_Toc17248)

[四、经验与教训 9](#_Toc16558)

**项目开发总结报告**

# 一、引言

## 1.1编写目的

【目的】：

a. 描述设计思路及过程，便于后期了解分析维护软件

b．对软件的适用性，可靠性以及可维护性进行评价分析。

【适用读者】：

软件客户、软件需求分析人员、软件开发人员、软件维护人员

## 1.2背景

在当今的校园生活中，越来越少的学生选择阅读纸质书。信息化电子化的产品不断的涌现出来，各种技术日新月异，的确为人们的生活提供了许多便利，但不可否认的是，纸质书籍仍有着它无可替代的作用和独特魅力。考虑到传统校园中图书馆的一些弊端，例如，书籍更新缓慢，多数书籍十分陈旧，书籍流动率不高，以及无法调动学生们的阅读热情等，该图书交换平台需求应运而生。

该平台借助现代互联网技术，实现线上的图书交换，让学生们将手中空闲的书籍分享出来，并在平台中借阅想要阅读的书籍，这种方式将传统图书与先进技术结合，能够极大的调动学生们的阅读热情，甚至以“读书社交”的方式，形成阅读圈子，广结好友，分享讨论，让纸质书籍借助现代技术继续展露她的魅力。

【软件名称】：“贝壳书吧”图书漂流微信小程序

【项目名称】：软件工程开发实践——微信小程序设计与开发

【项目提出者】：软件工程课程小组

【开发者】：雷镇宇，廖舒祺，李策，周尊康，迟韶辉，李笑林

【目标用户】：校园中热爱阅读的同学们

【软件运行平台】：基于微信的微信小程序平台

## 1.3定义

本软件所用到的专业名词术语如下：

Wafer2: 腾讯云下一代小程序综合解决方案

Koa2：基于 Node.js 平台的下一代 web 开发框架

WeUI：一套同微信原生视觉体验一致的基础样式库

## 1.4参考资料

微信小程序开发文档（https://developers.weixin.qq.com/miniprogram/dev/index.html）

Koa2官方文档（https://koa.bootcss.com/）

Wafer2使用文档

Knex.js官方文档（https://knexjs.org/）

WeUI官方文档（https://weui.io/）

# 二、实际开发结果

## 2.1产品

该平台的定位是线上的图书交换平台。拥有友好的UI界面，可以实现线上图书交换，进行图书分享和借阅等功能

项目文件：

|  |  |
| --- | --- |
| Client | Server |
| Components:  comment  item  rate  line  Pages:  aboutMe  bookDetails  bookList  bookReviews  borrowBook  index  manageBorrow  me  process  searchResult  shareBook  uploadAct  verifyBook  verifyName  verifyUser  Utils:  utils.js | routes:  index.js  controllers:  backBook..js  booksReadyForVerify.js  borrowBook.js  deleteAct.js  getBackBook.js  getBanner.js  getBookDetails.js  getBorrowBook.js  getBorrowDetails.js  getComments.js  getLike.js  getLikeBook.js  getNewBooks.js  getProcess.js  getShareBook.js  getUserMoreInfo.js  index.js  isAdmin.js  login.js  manageBorrow.js  notPassBook.js  notPassBorrow.js  notPassUser.js  passBackBook.js  passBookVerify.js  passBorrow.js  passUserVerify.js  recommend.js  returnBook.js  searchBook.js  upload.js  uploadAct.js  uploadBookInfo.js  uploadComment.js  uploadLike,js  uploadVerifyName.js  user.js  userInfo.js  usersReadyForVerify.js |

数据库：

1.admin

|  |  |
| --- | --- |
| **字段** | **含义** |
| aid | 管理员id |

2.banner

|  |  |
| --- | --- |
| **字段** | **含义** |
| bnid | banner id |
| title | banner标题 |
| intro | banner介绍 |
| image | banner图片 |

3.book

|  |  |
| --- | --- |
| **字段** | **含义** |
| bid | 图书id |
| uid | 分享者id |
| image | 图书图片 |
| bname | 图书名字 |
| author | 图书作者 |
| publish | 出版社 |
| price | 价格 |
| shell | 贝壳数 |
| ISBN | ISBN |
| intro | 介绍 |
| rate | 评分 |
| status | 状态 |
| date | 分享日期 |

4.borrow

|  |  |
| --- | --- |
| **字段** | **含义** |
| brid | 借书id |
| uid | 借书者id |
| bid | 图书id |
| borrow\_date | 借书日期 |
| return\_date | 归还日期 |
| status | 状态 |

5.comment

|  |  |
| --- | --- |
| **字段** | **含义** |
| uid | 评论用户id |
| bid | 评论图书id |
| date | 评论日期 |
| rate | 评分 |
| content | 评论内容 |

6.like

|  |  |
| --- | --- |
| **字段** | **含义** |
| uid | 收藏用户id |
| bid | 收藏图书id |

7.process

|  |  |
| --- | --- |
| **字段** | **含义** |
| pid | 管理员id |
| uid | 用户id |
| bid | 图书id |
| brid | 借书id |
| type | 类型，分享或借书 |
| date | 流程记录时间 |
| info | 流程信息 |

8.user

|  |  |
| --- | --- |
| **字段** | **含义** |
| uid | 用户id |
| open\_id | 微信内部用户唯一识别码 |
| nickname | 微信昵称 |
| avatar\_url | 微信头像 |
| gender | 性别 |
| schid | 学号 |
| name | 真实名字 |
| phone | 手机 |
| cardimg | 校园卡图片 |
| money | 贝壳 |
| credit | 信誉值 |
| isVerified | 是否实名认证 |

## 2.2主要功能和性能

该平台的定位是线上的图书交换平台。

### 2.2.1功能划分

图书分享模块，图书借阅模块，图书查阅模块，图书搜索模块，图书推荐模块，图书收藏模块，图书评论模块，货币系统模块，用户认证模块，管理员模块。

### 2.2.2图书分享

用户可以上传自己手中闲置的图书，通过扫描图书条形码获取图书ISBN，再通过豆瓣API获取图书信息，上传图书的基本信息，并将图书交付管理员，待管理员审核通过后，该图书的信息就发布成功并可以被其他用户看到。

### 2.2.3图书借阅

用户可以凭借手中的贝壳，在平台中寻找自己喜爱的图书，并进行借阅。进行借阅后，会生成一个借阅订单，进入下单状态，并扣除相应的贝壳。用户可以从管理员处领取书籍，并由管理员操作，完成订单。

### 2.2.4图书查阅

用户可以在图书详情页面查看图书信息，了解图书作者、定价、贝壳、评分以及用户评论等信息。

### 2.2.5图书搜索

用户可以在平台中搜索自己想要借阅的图书。

### 2.2.6图书推荐

系统采用基于用户的协同推荐算法，会根据用户的借阅、收藏喜好等信息，为用户推荐用户最可能想要阅读的书籍信息，做到推荐智能化，服务人性化。

### 2.2.7图书收藏

若用户发现自己想要借阅的图书已经被借阅，或者目前没有时间借阅，想要以后进行借阅，可以将图书加入自己的收藏中，方便查找。

### 2.2.8图书评论

每一个借阅过该图书的用户，均有权利对这本图书进行评价和评级，级别为一到五星。每本图书的星级为综合所有用户的评级计算得出。

### 2.2.9货币系统

系统内采用专用货币“贝壳”。在初始状态，系统中是没有贝壳的，贝壳作为货币，来源于实体的价值，在这里就是用户上传的书籍的价值。当用户上传书籍并被管理员验证通过后，可以获得一定数量的贝壳，这个数量的计算方式以书籍本身的定价为基准，由其新旧程度等因素上下浮动。

### 2.2.10用户认证

考虑到实际情况，该平台的服务目前只对校园内用户开放，因此尽管所有微信用户均可进入该平台，但只有通过实名认证的用户才可以使用该平台。平台认证条件即是提供姓名、学号、校园卡等信息，待管理员后台确认后方可完成认证。

### 2.2.11管理员

管理员可以进行系统管理。管理员可以进行用户审核、图书管理、借阅管理、发布和下架活动信息等功能。

## 2.3进度

可行性研究

分析使用开发语言以及开发框架进行编写更加快速便捷，效率也比较高。以及开发所需要的时间。

需求分析

分析该微信小程序需要完成那些功能，实现那些效果，以及用户期待的体验

总体设计

设计该微信小程序的总体架构，数据库的使用，前后端的交互以及UI设计

详细设计

详细设计微信小程序每张页面的具体功能和路由逻辑，以及需要的接口API和算法实现

单元测试

单元测试每个模块是否完成已有的功能，是否程序中还存在BUG等等

综合测试

综合测试所有的功能

# 三、开发工作评价

## 3.1对生产效率的评价

生产效率十分高效。团队分工明确，团结合作，在UI设计，数据库建立，前后端代码编写以及BUG测试和修改的过程中展现团队的力量，按期完成所有工作。

## 3.2对产品质量的评价

产品质量较高，完成了预计的功能具有较高的容错性。

## 3.3对技术方法的评价

**1.开发技术：**

**基于『Koa2』Node.js框架的『Wafer2』微信小程序开发框架**

**评价：**使用官方推荐的Wafer2开发小程序，搭配腾讯云服务器使得开发环境的搭建十分便利，能够快速的投入开发工作中，并且能够借助Wafer2提供的基础API实现一些较为复杂的功能。

**数据库存储**

**评价：**使用MySQL数据库，并使用Knex.js来操作数据库，其最大优点在于实现前后端以及数据库统一使用Javascript作为开发语言，书写更加统一，交互更加灵活。

**2.开发工具：**使用微信官方推出的微信Web开发工具，开发十分方便。其搭载的腾讯云远程调试、预览、测试等功能使开发更加便捷。

## 3.4开发中出错原因的分析

首页“猜你喜欢”板块报错：

原因：微信小程序启动生命周期顺序导致了加载该板块时没有获取用户的uid，无法进行智能推荐

解决方法：考虑到实际情况，新用户无法进行只能推荐，因此新用户的推荐内容为热门图书，由于移动端会有缓存，当用户再次进入时会获取到uid进行只能推荐

Wafer2中提供的图片上传API无法正常使用：

原因：官方的SDK库更新导致API版本不一致

解决方法：重新部署代码，使用最新SDK

豆瓣API无法使用：

原因：未知。猜测可能豆瓣封禁了来自微信小程序的请求

解决方法：借助他人反代理的服务器，间接实现豆瓣API的功能

# 四、经验与教训

首先，本项目的结构非常复杂，源文件众多。由于在最初对于微信小程序开发并无经验，我们阅读了大量官方文档以及Wafer2和Koa2的基本原理，和架构组成，慢慢对微信小程序的开发有了一定掌握，边开发边进步，最终将成本制作出来。

通过该项目，我们更加深刻的理解了软件工程的理论知识和实际应用，同时对微信小程序的开发有了更深的认识，积累了大量工程经验。最重要的是，我们在软件开发的过程中，经历了提出问题、可行性分析、需求分析等一系列步骤，经历了一个工程从无到有，从简陋到复杂，从只有一个界面到可以实现发送文件的蜕变，这对我们深入理解软件工程这门课，进而深入理解软件开发这项工作是极为有益的。同时，在软件开发的过程中，我们也遇到许多困难和问题，但通过不懈努力和组员之间的相互帮助，都一一克服，最终获得一个较为完美的成果，尽管它仍然有许多不足和缺陷，但我们已经收获了很多。

通过本次开发，我们提高了对软件工程这门课的认识，加强了自己的工程能力，为以后从事软件开发工作打下了基础。