**图书借阅系统**

**需求分析说明书**

目录

[1.引言 1](#_Toc480018369)

[1.1 目的 1](#_Toc480018370)

[1.2 范围 1](#_Toc480018371)

[1.3 参考文献 1](#_Toc480018372)

[2.总体概述 2](#_Toc480018373)

[2.1 开发背景 2](#_Toc480018374)

[2.2 商品功能 2](#_Toc480018375)

[2.3 约束 4](#_Toc480018376)

[2.4 假设和依赖 4](#_Toc480018377)

[3.详细需求描述 5](#_Toc480018378)

[3.1 对外接口需求 5](#_Toc480018379)

[3.1.1 用户界面 5](#_Toc480018380)

[3.1.2 通信接口 10](#_Toc480018381)

[3.2 功能需求 10](#_Toc480018382)

[3.2.1 系统角色 10](#_Toc480018383)

[3.2.2 普通用户功能需求 12](#_Toc480018384)

[3.2.2.1 用户注册 12](#_Toc480018385)

[3.2.2.2 用户登录 13](#_Toc480018386)

[3.2.2.3 切换用户 13](#_Toc480018387)

[3.2.2.4 搜索图书 13](#_Toc480018388)

[3.2.2.5 查看图书详情 14](#_Toc480018389)

[3.2.2.6 预定书籍 14](#_Toc480018390)

[3.2.2.7 借书 15](#_Toc480018391)

[3.2.2.8 还书 15](#_Toc480018392)

[3.2.3 管理员用户功能需求 15](#_Toc480018393)

[3.2.3.1 注册 15](#_Toc480018394)

[3.2.3.2 登录 16](#_Toc480018395)

[3.2.3.3 管理借书 16](#_Toc480018396)

[3.2.3.4 管理还书 17](#_Toc480018397)

[3.3 非功能性需求 17](#_Toc480018398)

[3.3.1 性能需求 17](#_Toc480018399)

[3.3.2 安全性需求 17](#_Toc480018400)

[3.3.3 可用性需求 18](#_Toc480018401)

# 1.引言

## 1.1 目的

本文档描述了图书借阅软件的功能需求和非功能性需求，并试图从整体架构上给出整个系统的轮廓，对软件的功能需求、性能需求和其他非功能性需求给出了详细的描述，并对软件功能部分使用了系统用例进行了描述。

该文档详细说明了软件的需求和规格，这些说明是软件进行设计和开发的基础，是开发者必须明确的问题，这也是编写测试用例和进行系统测试的主要依据。

除特殊说明之外，本文档所包含的所有需求都是高优先级需求。

## 1.2 范围

信息时代的到来，给我们带来了一系列的便利，正如我们所享受到的，买东西再也不用出门，打开手机，你就可以完成订单，这一系列的便利当然离不开互联网的发展，当然我们更要感谢给我们开发出便利软件的公司，像阿里、腾讯等互联网公司，正是因为他们的发展，给我们的生活带来了变化。

本次图书借阅软件的开发，正是要基于微信开发，微信给我们带来了生活上的便利，当然对于我们开发人员来说，它还带给我们了开发的便利。本次我们要基于微信开发一款微信小程序，目标是给我们现在的图书馆带来便利和爱看书的朋友带来方便，通过这款软件，图书馆可以更方便的管理图书，借书的朋友也可以更加方便的去借阅图书。

## 1.3 参考文献

软件架构设计实践教程 张家浩编著 北京:清华大学出版社

软件工程模型与方法.第2版 肖丁, 修佳鹏编著 北京邮电大学出版社

软件系统架构与开发环境 郑建德编著 机械工业出版社

Android产品实战从零开始 黄宇健编著 清华大学出版社

软件测试第二版 佟伟光编著 人民邮电出版社

# 2.总体概述

## 2.1 开发背景

根据我们团队进行的市场调查，大多数人的需求是看是而不是买书，有很大一部分人买完书阅读后就不在仔细阅读导致书的可读性和价值降低。

比如有的人想看书，却画大价钱去买，而买完后却束之高阁，从来不去过问，正所谓“书非借不能读也”，大部分的我们都是这样，买完了书，看不了几页，就扔在一旁，再也不去过问，而解决这个问题最好的方法就是去图书馆借书。

但是在借书的过程中还会出现一些问题，比如当查阅图书馆的网站查看后，过了几天，当你去借书的时候，发现书已经被借走了，再比如去图书馆借书发现要办理各种卡，然后需要交付一定的押金，还有借书之后你需要自己记住归还的日期，我们往往会忘记归还而面临扣费等。

本次系统的开发，就是朝着解决这些问题去的，我们将利用微信这个平台让大多数喜欢阅读的朋友得到更好的便利，有了这个系统我们可以直接在这个平台上进行预定，借阅，并且这个系统将定期给我们提醒该归还的图书。

## 2.2 商品功能

功能列表

|  |  |
| --- | --- |
| 功能编号 | 功能简述 |
| SF1 用户注册 | 用户可以通过自己的手机号或者微信ＩＤ进行注册。 |
| SF2 用户登录 | 用户通过账号、密码登录系统 |
| SF3 找回密码 | 用户通过提供绑定的手机号，获取验证码，重新设置密码来找回密码 |
| SF4 图书导航 | 应用界面按照不同学科类型的图书进行导航，并可以给出各个学科中代表性的图书供用户参考。 |
| SF5 搜索图书 | 用户通过关键字、拼音全拼或者首字母、图书编号等搜索书籍。将搜索到的图书以列表的形式进行展示。列表的内容包括缩图、书名、作者、藏书量、可借阅量。 |
| SF6 扫码功能 | 用户使用该小程序的扫码界面，扫描图书的二维码，获得图书的详细信息，如书名、出版社、作者、藏书量，可借阅量、图书的缩图、图书在图书馆的收藏位置等一系列的信息。 |
| SF7 展示图书详情 | 用户从搜索列表中点击图书，进入到图书详情中或者通过扫描图书二维码进入到图书详情页，会显示这本书的详细信息，包括书籍的标题、出版社、版本号、封面、序、目录、内容简介、书评和导读、藏书量等信息。 |
| SF8 推荐相关书籍 | 用户在查看一本书的时候，在这本书详细内容介绍的页面上，我们会给给用户提供一些相关的书籍，当用户点击相关书籍的时候，会自动跳转到该书籍的相对的详细内容介绍的界面。 |
| SF9 推荐阅读 | 系统会根据用户平时的浏览记录，定期推送推荐图书，当然用户可以设置推荐频率或者关闭推荐功能。 |
| SF10 预定图书 | 用户注册登录之后在图书详情页，可以进行预定，系统默认预定时间为7天，超过时间，系统将自动取消用户的预定。如果不巧没有藏书，用户可以选择当该图书被归还的时候，系统自动发送推送信息，告诉用户该图书归还，可以去借书了。 |
| SF11 借阅图书 | 用户通过点击图书详情页的加入借书栏按钮，然后生成用户独有的借书二维码，管理员使用管理app扫描用户出示的借书二维码，用户支付押金成功之后，代表用户成功借书。 |
| SF12 还书提醒 | 一本书借出去有还书提醒，系统从倒计时1周开始给用户推送还书提醒，如果用户过期没有归还图书，则将发送过期未还提醒。 |
| SF13 还书 | 用户携带书籍去图书馆，出示自己的借书二维码给管理员。管理员通过管理app扫描用户出示的二维码得出借书清单。与实物比较无误后办理书籍入库，完成还书环节。 |
| SF14 查看借阅历史 | 用户可以在查看所有借阅信息，如图书编号，借书日期，馆藏地址，还书日期等信息。 |
| SF15 续借 | 每本借阅的图书用户可以续借1次，续借期限为1个月。 |
| SF16 加入借书栏 | 在图书详情页，用户可以点击“加入借书栏”按钮，将图书添加到借书栏中，表示用户想要借阅该图书。 |
| SF17 加入预定列表 | 在图书详情页，用户可以点击“加入预定列表”按钮，将图书添加到预定列表中，表示用户想要预定该图书。 |
| SF18 取消预定 | 在预定列表页中，用户在通过点击预定项上的“取消预定”按钮，可以针对某一本书取消预定，在预定列表中，将其删除。 |
| SF19 取消借阅 | 在借书栏页中，用户通过长按某一本书，可以选择在借书栏中删除此书，若用户在弹出的对话框中选择“确定”，则代表用户取消借阅了该书。 |

## 2.3 约束

约束依赖表

|  |  |
| --- | --- |
| 约束编号 | 约束简介 |
| CON1 | 系统将运行在移动终端上，支持android系统。 |
| CON2 | 系统是基于微信开发的，上架运行需经过腾讯审核。 |
| CON3 | 开发的框架是微信提供给我们的开发框架。 |
| CON4 | 在开发过程中，开发者要提交软件需求规格说明书，软件概要设计文档。 |

## 2.4 假设和依赖

|  |  |
| --- | --- |
| 假设依赖编号 | 假设依赖表 |
| AE1 | 因为服务器条件限制，软件同时使用人数不超过100人。 |
| AE2 | 同一账户不能在多个终端上登录。 |
| AE3 | 后台图书数据均是模拟数据，没有连接实际的的图书馆的书籍数据库。 |

# 3.详细需求描述

## 3.1 对外接口需求

### 3.1.1 用户界面

**UI 01 登录界面**

登录界面分为账号登录和短信快捷登录或者第三方微信登录。

界面如下：

图3.1.1-1 图3.1.1-2

**UI 02 注册界面**

用户填写手机号，添加身份证号，真实姓名，设置密码注册新账号。

界面如下：



图3.1.1-3

**UI 03图书列表形式展示界面**

当我们按照某一种搜索要求搜索图书的时候，会将搜索出的图书结果展示在下拉列表中，供借阅者参考。点击下拉列表中的某一项会跳转到相应的图书详情页中，如果点击的放大镜按钮将会进入到搜索结果的详细页中。

界面图示为：

图3.1.1-3.1 图3.1.1-3.2

**UI 03 图书详情页**

该界面是图书信息的详细信息界面，主要包括书目信息（包括书名、ISBN、作者、出版社、目录、序言、导读）、馆藏地 书刊状态等具体信息，用户预定列表超过5本或者借书栏中超过8本会给出相应的提示。

界面图示为：

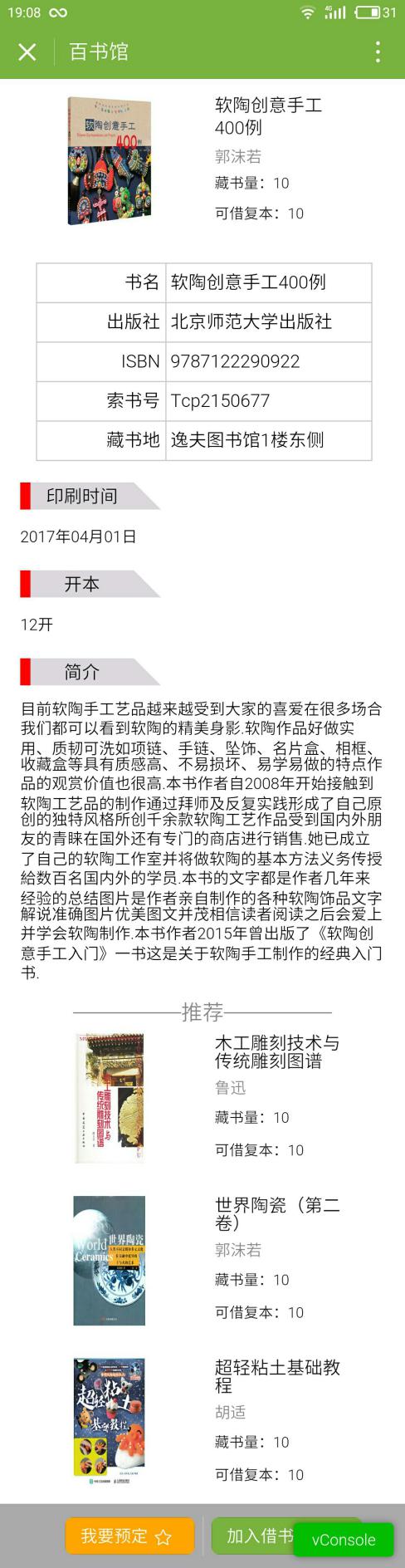


图3.1.1-3

**UI 04 首页**

首页设计比较简单，也便于用户操作。上面时搜索部分，用户可以直接输入图书信息等进行搜索，或者用扫码功能进行搜索，再下面一部分是图书的分类，然后是图书的推荐，最低端部分是一个Tab选项卡，通过点击不同的选项跳转到不同的功能界面。

界面图示为：



图3.1.1-4

**UI 05 借书栏界面**

本界面主要显示用户想要借阅的图书列表，每一项包括图书的基本信息，如编号、图书封面图等，点击图片跳转到图书详情展示页（点击《十日谈》跳转到图书详情），长按图片可以删除该本图书。最下面是一个按钮，用户选中要借阅的图书然后点击该按钮可以生成借书的二维码。

界面图示为：

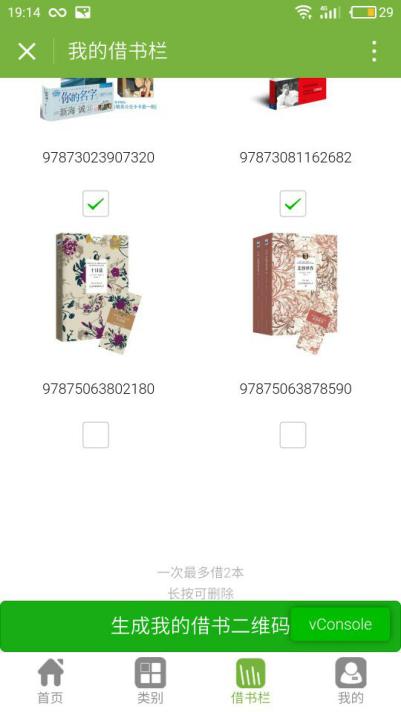
 

图3.1.1-5.1 图3.1.1-5.2

图3.1.1-5.3 图3.1.1-5.4

**UI 06 借书二维码界面**

此界面只需要简单的一个二维码，不需要别的修饰。

界面图示为：



图3.1.1-6

**UI 07 个人中心界面**

本界面是显示用户信息以及用户操作信息的界面，包括用户名，账号，以及借阅历史，借书栏，预定列表、设置、退出登录等基本信息，用户点击相应的链接，会跳转到相应的界面，展示具体信息。本界面设计简单，便于用户操作，用户方便易懂。

界面图示为：



图3.1.1-7

**UI 08 用户登录界面**

本界面包含账户输入，密码输入，登录按钮等基本控件。当然用户忘记密码的时候，可以在该界面直接点击忘记密码按钮，进行相应操作，未注册过的用户也可以直接点击注册按钮，跳转到注册界面进行相应的注册操作，如果用户不想用此方法登录，也可以选择短信快捷登录。

界面图示为：

图3.1.1-8.1 图3.1.1-8.2

**UI 09 找回密码**

用户忘记密码，通过该页面找回密码，输入手机号，获取验证码

界面图示如下：

图3.1.1-9.1 图3.1.1-9.2

**UI 10借阅历史**

这个界面将会记住用户在过去借阅并且归还的图书信息。

界面图示为：



图3.1.1-10

**UI 11 还书**

在这个界面的中用户可以选择自己将要归还的书籍，点击图片跳转到图书详情，选中要还的书籍，点击生成还书二维码，供管理员app扫描，管理员与实物比较无误后，用户就完成了还书操作。

界面如下：

图3.1.1-11.1 图3.1.1-11.2



图3.1.1-11.3

**UI 12 预定列表界面**

在图书详情页中，点击我要预定按钮，在符合条件的情况下，可将该书加入预定列表中。 在预定列表界面中用户可以使用这个界面查看预定的图书，用户可在这个界面中进行取消借阅或者加入借书栏的操作。

界面如下：

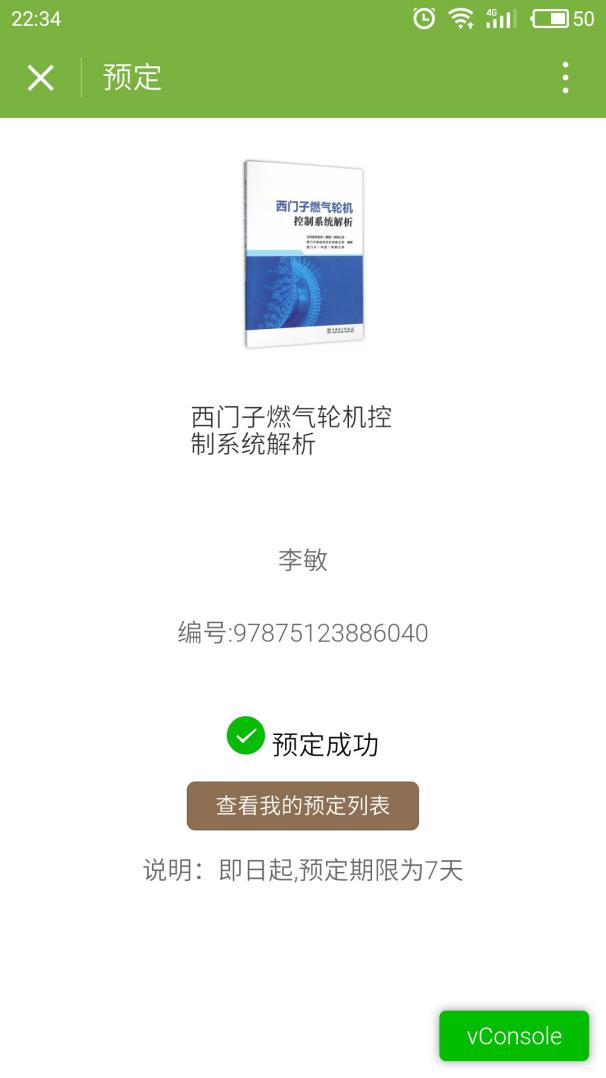
 

图3.1.1-12.1 图3.1.1-12.2

图3.1.1-12.3 图3.1.1-12.4



图3.1.1-12.5

**UI 13 用户设置界面**

用户在这个界面中可以设置是否接收系统发出的提醒（取书和还书），还可以设置用户是否接受推荐书籍和设置推荐书籍的频率（一周一次或者一月一次）。

界面如下：

图3.1.1-13.1 图3.1.1-13.2



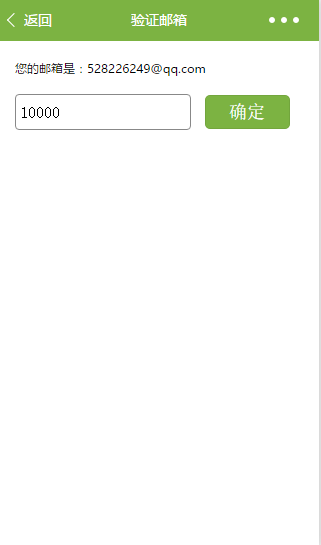
图3.1.1-13.3

**UI 14 绑定邮箱**

用户提交自己的邮箱地址，验证了邮箱之后，可以接受系统发送的提醒和图书推荐。

界面如下：

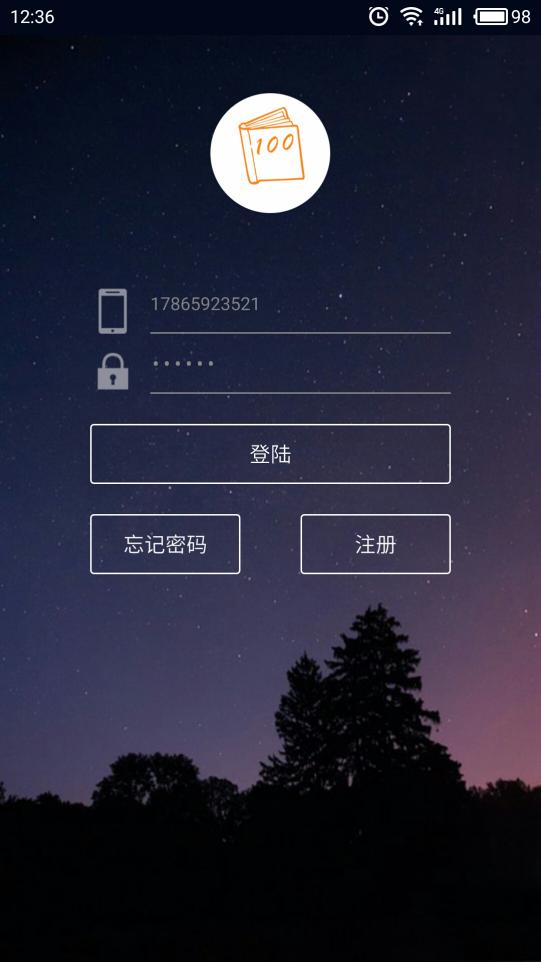
UI 15 支付



### 3.1.2 管理员界面

**UI 01 登录界面**

管理员输入手机号、密码，系统验证正确之后，登录系统。界面如下：

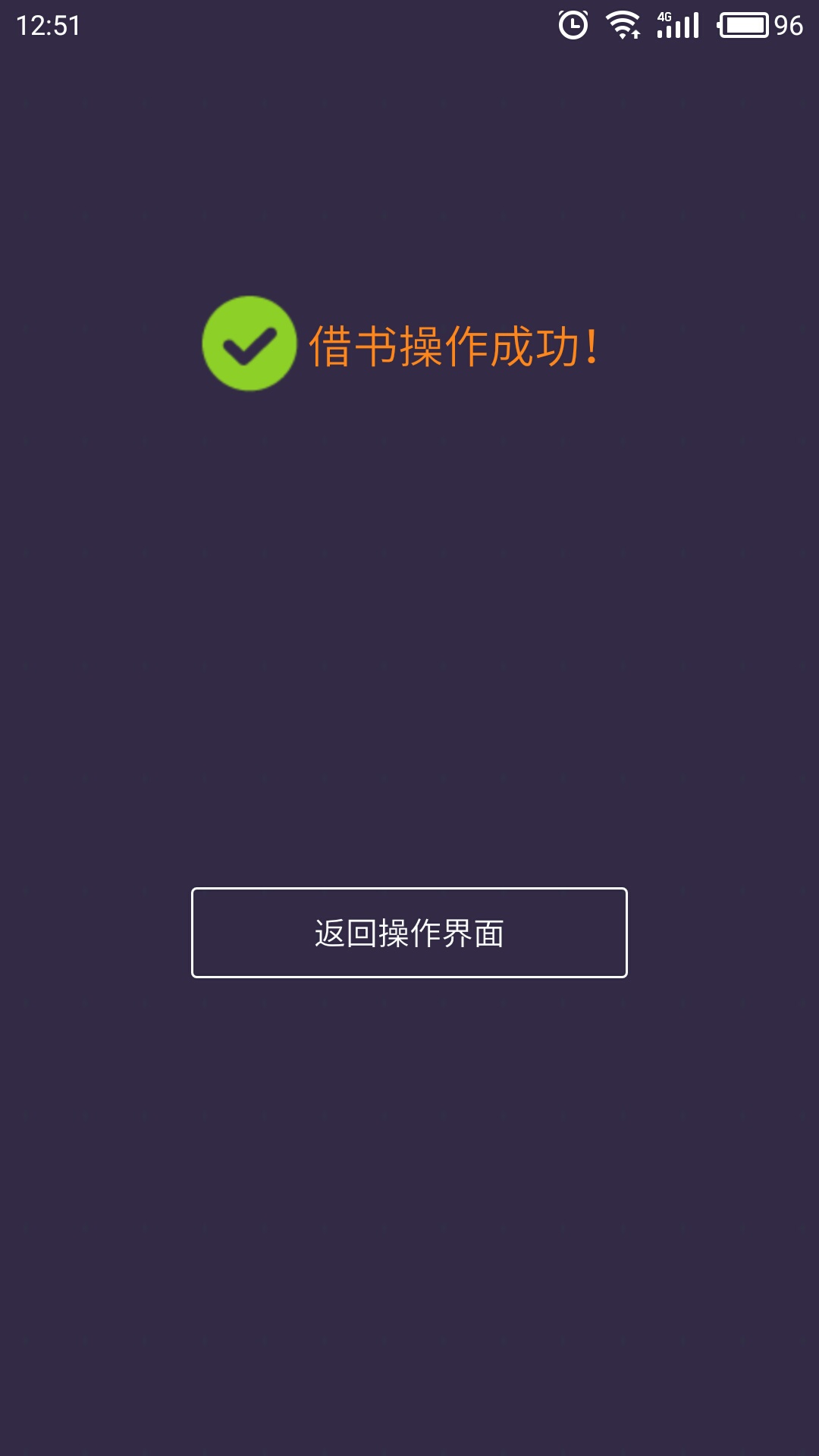
**UI 02 注册**

**用户输入手机号获取验证码，输入正确的验证码之后设置密码。界面如下：**

** **

**UI 03 借书**

** **

****

**UI 04 还书**

** **

****

UI 05 借书记录

### 3.1.3 通信接口

因为我们的服务器端采用java语言编写，而客户端使用的是类似于HTML5的一种微信提供的开发环境，因此服务器端与客户端使用的语言不同，我们要跨语言进行通信，所以，我们使用RPC的方式进行通信。

## 3.2 功能需求

### 3.2.1 系统角色

本系统主要用于以下几类人员：

（1）未登录注册用户

在本系统未进行登录操作，只享有本系统的部分功能，例如：搜索书籍，查看本站推荐的书籍等一些功能。

用例图如下：

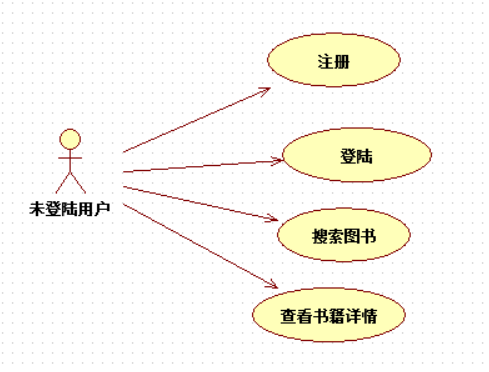


图3.2.1-1

（2）已登录用户

该类用户除了不享受管理员功能之外，是本系统的最高权限使用者，享有本系统的一切功能，可以查看书籍详情，搜索书籍，借书，还书，预定图书等一系列的操作。

用例图如下：

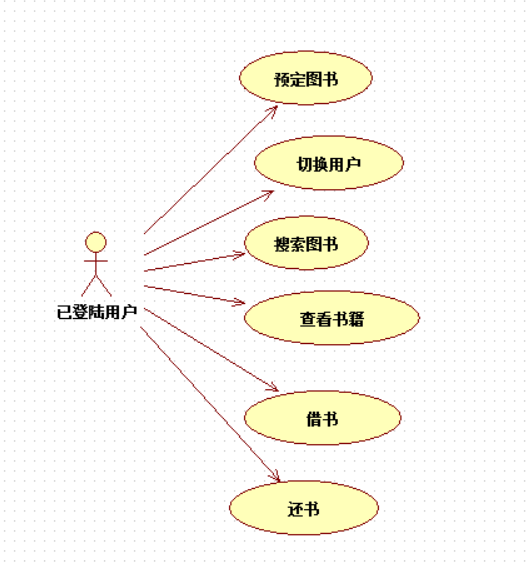


图3.2.1-2

1. 管理员

该类用户通过是管理图书借还的管理者，通过他们的同意才能将图书正常的借出与正常的归还。该类用户必须严格登录注册后才能使用，他们管理借书与还书。

用例图如下：

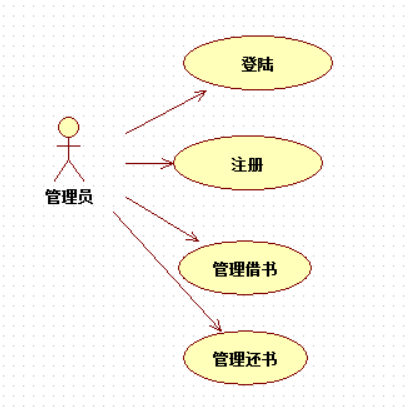


图3.2.1-3

### 3.2.2 普通用户功能需求

#### 3.2.2.1 用户注册

|  |  |
| --- | --- |
| 使用者 | 所有访问该系统的用户。 |
| 目的 | 明确每一个使用者，便于系统管理。 |
| 基本事件流 | 1.用户进入登录界面，如若没有注册过，则点击注册新用户，进入注册界面，本用例开始。  2.注册界面显示需要用户填写的基本注册信息，如：微信号，手机号，真实姓名，身份证号，短信验证码等。  3.用户填写完成并填写正确，点击提交。  4.系统检查用户输入信息的正确性，如果正确，则注册成功，本用例借书，否则提示错误信息，请用户重新输入。 |

#### 3.2.2.2 用户登录

|  |  |
| --- | --- |
| 使用者 | 已经注册过的用户。 |
| 目的 | 识别身份信息。 |
| 基本事件流 | 1.用户点击我的图书，进入登录界面，本用例开始  2.系统显示请输入微信号或者手机号，请输入密码  3.用户填写上述提示内容，点击提交  4.系统检测用户输入信息的正确性，如若正确，则开启我的图书馆界面，本用例结束，如若不正确提示错误信息，请用户重新填写并提交。 |

#### 3.2.2.3 切换用户

|  |  |
| --- | --- |
| 使用者 | 已登录的用户。 |
| 目的 | 便于多个用户使用一台终端进入系统操作。 |
| 基本事件流 | 1.用户点击切换用户，本用例开始。  2.系统提示是否切换用户，如果用户点击是，则进入登录界面，如果点击否，则本用例结束。 |

#### 3.2.2.4 搜索图书

|  |  |
| --- | --- |
| 使用者 | 所有进入系统的用户。 |
| 目的 | 检索用户想要了解的图书。 |
| 基本事件流 | 1.用户在检索框中输入信息，点击搜索，或者用扫描图书二维码，本用例开始。  2.系统根据用户输入的信息开始进行检索，并将检索的内容显示在图书列表页，本用例结束，如果搜索的内容不存在，则系统提示用户搜索的内容不存在，本用例结束。 |

#### 3.2.2.5 查看图书详情

|  |  |
| --- | --- |
| 使用者 | 所用进入系统的用户。 |
| 目的 | 使用户更加了解图书信息。 |
| 基本事件流 | 1.用户在图书列表页点击一种图书，或者在推荐图书中点击推荐的一本图书，即进入查看图书详情页，用例开始。  2.系统将根据用户点击的图书，展示出图书的详细内容，本用例结束。 |

#### 3.2.2.6 预定书籍

|  |  |
| --- | --- |
| 使用者 | 已登录的用户。 |
| 目的 | 用户通过预定书籍，可以在一定的时间返回内，预定自己想要借的书籍，而防止被别人提前借走。 |
| 基本事件流 | 1.用户在图书详细页点击“我要预定”按钮，则本用例开始。  2.系统将本书加入用户预定书籍栏，并提醒用户在多长时间之前去借阅，否则将被退回。  3.系统将本书的状态改为已预定，不可被别的用户预定或借阅。 |

#### 3.2.2.7 借书

|  |  |
| --- | --- |
| 使用者 | 已登录的用户。 |
| 目的 | 用户可以通过该系统办理借书手续。 |
| 基本事件流 | 1.用户打开扫码功能，本用例开始。  2.用户扫描书上的二维码，系统识别二维码，并提取该书信息。  3.系统弹出是否借阅按钮，如果用户点击是，则将该书加入借书栏，并生成相应借书单二维码。，如果用户点击否，则本用例结束。  4.用户打开自己的借书单上的二维码，管理员进行扫描，验证通过，则用户缴纳一定的押金，借书成功，系统修改书籍借阅状态和用户自己的个人图书馆中的信息，本用例结束。 |

#### 3.2.2.8 还书

|  |  |
| --- | --- |
| 使用者 | 在图书馆借过书的用户。 |
| 目的 | 用户通过该系统才能正常还书。 |
| 基本事件流 | 1.用户可以通过出示自己的借书单的二维码，管理员扫码成功后，与书籍进行比对，书籍入馆，系统将自行修改用户个人图书馆中的图书信息，本用例结束。  2.用户还可以直接出示自己借的书籍，管理员通过扫描书籍上的二维码，将书籍入馆，系统也将修改借书用户的个人借书信息，本用例借书。 |

#### 3.2.2.9 支付押金

|  |  |
| --- | --- |
| 使用者 | 借阅图书的用户。 |
| 目的 | 用户通过该系统才能正常借书。 |
| 基本事件流 | 1.管理员通过管理app扫描用户的借书二维码，与实物比较无误后，点击循序借阅按钮，本用例开始。  2.系统自动跳转到支付界面，用户在该页对比书籍信息以及个人信息，输入支付密码，支付成功，本用例结束。 |

#### 3.2.2.10 设置

|  |  |
| --- | --- |
| 使用者 | 已登录用户。 |
| 目的 | 设置是否接受系统推荐。 |
| 基本事件流 | 1. 用户在“我的”界面，点击设置，本用例开始。 2. 用户可以选择是否接受提醒（取书、还书），设置推荐书籍的频率（一周一次或一月一次），或者不接受推荐书籍。 3. 用户设置完毕，本用例结束。 |

### 3.2.3 管理员用户功能需求

#### 3.2.3.1 注册

|  |  |
| --- | --- |
| 使用者 | 图书馆的相关工作人员。 |
| 目的 | 图书馆的相关工作人员同意注册后，才能注册管理员。 |
| 基本事件流 | 1.用户点击注册，进入注册界面，用例开始  2.系统展示基本注册信息，手机号，真实姓名，身份证号，注册码（注意：注册码为图书馆所有者提供）。  3.用户根据提示填写信息，点击注册  4.系统根据所填信息进行检查，如果验证通过，则注册成功，本用例结束，如果未通过，则提示错误，请重新注册。 |

#### 3.2.3.2 登录

|  |  |
| --- | --- |
| 使用者 | 已经注册过的管理员。 |
| 目的 | 验证管理员的身份信息，保证系统安全性。 |
| 基本事件流 | 1.用户点击登录，用例开始。  2.系统提示填写员工号，密码。  3.用户根据提示填写信息，并点击提交，  4.系统验证用户提交信息，如果正确，则用例结束，如果错误，则提示错误，请重新填写。 |

#### 3.2.3.3 管理借书

|  |  |
| --- | --- |
| 使用者 | 已登录过的管理员。 |
| 目的 | 管理员通过后，方可将图书带离图书馆。 |
| 基本事件流 | 1.管理员扫描用户出示的借书单，扫描通过后，系统将发送请求给服务器。  2.服务器收到请求后，发送给借书用户交押金的提示，用户提交押金后将，结果反馈给服务器。  3.服务器接收到后，发送同意信息，则借书成功。  4.系统将修改图书借阅状态信息以及用户个人图书信息。 |

#### 3.2.3.4 管理还书

|  |  |
| --- | --- |
| 使用者 | 已登录过的管理员。 |
| 目的 | 管理员检验通过后，借阅的图书才能正常入馆。 |
| 基本事件流 | 1.管理员扫描用户出示的借书单，验证通过后，图书入馆。  2.系统修改图书借阅状态信息以及用户个人图书信息。 |

#### 3.2.3.5 查看借阅历史

|  |  |
| --- | --- |
| 使用者 | 已登录过的管理员。 |
| 目的 | 校验已处理的借书信息。 |
| 基本事件流 | 1. 在管理员主页点击“查看借书记录”按钮本用例开始。 2. 系统展示7天之内的所有借阅信息。 3. 用户关闭该页，本用例结束。 |

#### 3.2.3.6 查看还书历史

|  |  |
| --- | --- |
| 使用者 | 已登录过的管理员。 |
| 目的 | 校验已处理的还书信息。 |
| 基本事件流 | 1. 在管理员主页点击“查看还书记录”按钮本用例开始。 2. 系统展示7天之内的所有还书信息。 3. 用户关闭该页，本用例结束。 |

## 3.3 非功能性需求

### 3.3.1 性能需求

（1）客户端一般响应时间不超过3秒（网络信号差除外）。

（2）支持上百台客户端同时使用。

（3）支持多个管理员客户端同时处理借书与还书操作。

（4）用户扫码时，识别二维码的时间不超过3秒。

（5）用户搜索图书的时候，系统搜索时间不超过3秒。

### 3.3.2 安全性需求

（1）权限控制

根据不同用户角色，设置相应权限，用户的重要操作都做相应的日志记录以备查看，没有权限的用户禁止使用系统，管理员权限与普通用户权限分离清楚，不存在任何交集，未登录用户不能使用借书等一些操作。

（2）重要数据加密

本系统对用户密码等一些安全性数据进行简单加密，并需要真实身份认证，以及手机号绑定，确保用户信息安全。

（3）记录日志

本系统应该能够记录系统运行时所发生的所有错误，包括本机错误和网络错误。这些错误记录便于查找错误的原因。日志同时记录用户的关键性操作信息。

### 3.3.3 可用性需求

（1）方便操作，操作流程合理

本系统开发尽量从用户角度出发，考虑用户的感受，以方便用户使用。如：用户可以在不登录的情况下查询书籍，这就考虑了用户的方便性，当一部分用户只想浏览以下图书，而并非真正借阅的时候，就不必麻烦登录系统。

（2）一些重要操作会给用户友好提示

当用户借阅图书时，或者退出更换用户的时候，会提示用户是否进行该操作，如果用户不幸点错，也可以返回。

（3）可返回性

当用户操作错误，或者更改了想法，可以随时更改回原来的操作，如用户预定图书，或者借阅图书，发现这本书不想看了，可以随时从自己的信息单中删除。

（4）用户可自定义

本系统中会给用户定期发送一些图书推送，或者是还书提醒，这些信息的推送频率，用户可以自己定义。