



ReadTogether

软件设计文档

目 录

1 开发规划	1
1.1 开发人员	1
1.2 开发环境和工具	2
1.3 开发规范	2
2 总体设计	3
2.1 动作及概念术语描述	3
2.2 基本设计描述	4
2.3 系统类结构图、序列图与网页设计关系图	4
2.4 主要界面流程描述	5
2.4.1 注册界面流程	6
2.4.2 借书界面流程	7
2.4.3 榜单与评论功能	8
2.5 模块列表	9
3 数据设计	10
3.1 数据库表结构	10
3.2 数据库 ER 图	10
4 接口规范	11
4.1 图书模块	11
4.1.1 Book	11
4.1.2 BookCopy	11
4.2 用户模块	11
4.2.1 MyUser	11
4.3 借阅行为模块	11
4.3.1 Borrowing	11
4.4 评论模块	12
4.4.1 Comment	12
4.5 后台管理模块	12
4.5.1 Info	12
4.6 榜单模块	12
5 模块设计	13
5.1 图书模块设计	13
5.2 用户模块设计	14
5.3 借阅行为模块设计	14
5.4 评论模块设计	14
5.5 后台管理模块设计	15
5.6 榜单模块设计	15
6 附录	16
6.1 参考资料	16

1 开发规划

1.1 开发人员

姓名	主要职责	备注
武祥晋	■ 组长 ■ 前后端衔接	
涂珂	■ 数据库设计与管理	
傅左右	■ 页面设计	
秦同	■ 页面设计	
李响	■ 各种文档	
张思浩	■ 各种文档	

1.2 开发环境和工具

工具	作用
Git 1.7.9.5	开发辅助工具
Python 2.7.3	Web 页面， Web Service
Django 1.5.5	Web 页面， Web Service
MySQL 5.5.34	数据库
MySQLLdb 1.2.4	Python 访问数据库的接口

1.3 开发规范

文档名称	位置
Python 编码规范	

2 总体设计

2.1 动作及概念术语描述

2.1.1 续借

当用户希望延长借阅期限时可以在网站上申请续借，当且仅当该书未被预约。

2.1.2 预约

当用户需要借阅一本暂时被借完的书时，可以选择预约功能，系统在书有存量的时候按预约时间先后分配给已预约的用户。

2.1.3 借阅期限

每本书的借阅期限根据用户与书籍的属性分为 0、7、14 天三种。

2.1.4 黑名单

当用户逾期未归还书籍或遗失书籍未缴罚款时，将被系统自动加入黑名单中，黑名单用户只有搜索、还书操作，管理员也可以手动管理黑名单。

2.1.5 ISBN

ISBN 是国际通用的图书或独立的出版物（除定期出版的期刊）代码，一个国际标准书号只有一个或一份相应的出版物与之对应。新版本如果在原来旧版的基础上没有内容上太大的变动，在出版时也不会得到新的国际标准书号码。

2.1.6 馆藏位置

标志图书实体在图书馆中的位置，方便读者查找。

2.1.7 上架

在用户还书后工作人员将图书放至指定位置，并修改图书状态。

2.1.8 图书丢失

当图书丢失时，用户应缴纳相关罚款，管理员修改图书状态。

2.2 基本设计描述

ReadTogether 的设计采用了 django 框架，后端实现则由 python 完成，而前端以 html 和 css 以及少量的 js 代码实现。具体而言，本项目包括 4 部分：

1) server: 一个轻量级的 web 服务器，由于采用了 django 构架，该部分已实现。

2) controller: 提供前端与后端间的通信。

3) 前端: 通过 web 页面接收用户的请求，通过与后端的交互显示结果。

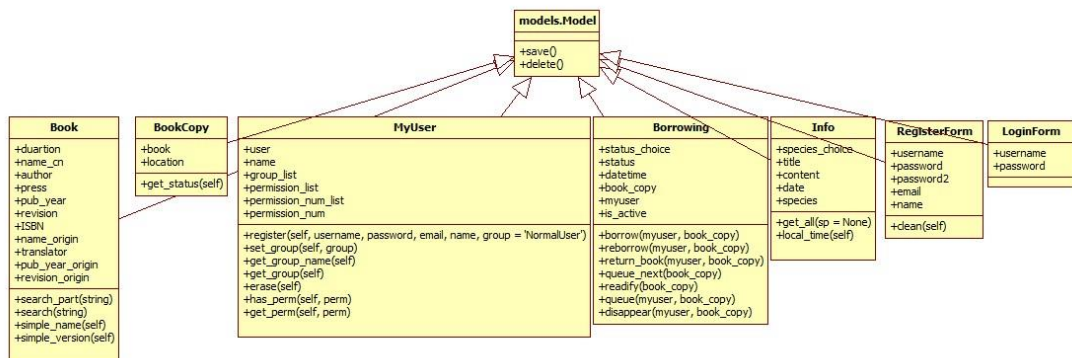
4) 后端: 通过对图书、用户、借阅行为等多个数据库的建立，实现数据的存储与查询等操作，并返回前端所需要的结果。

所涉及到的文件：

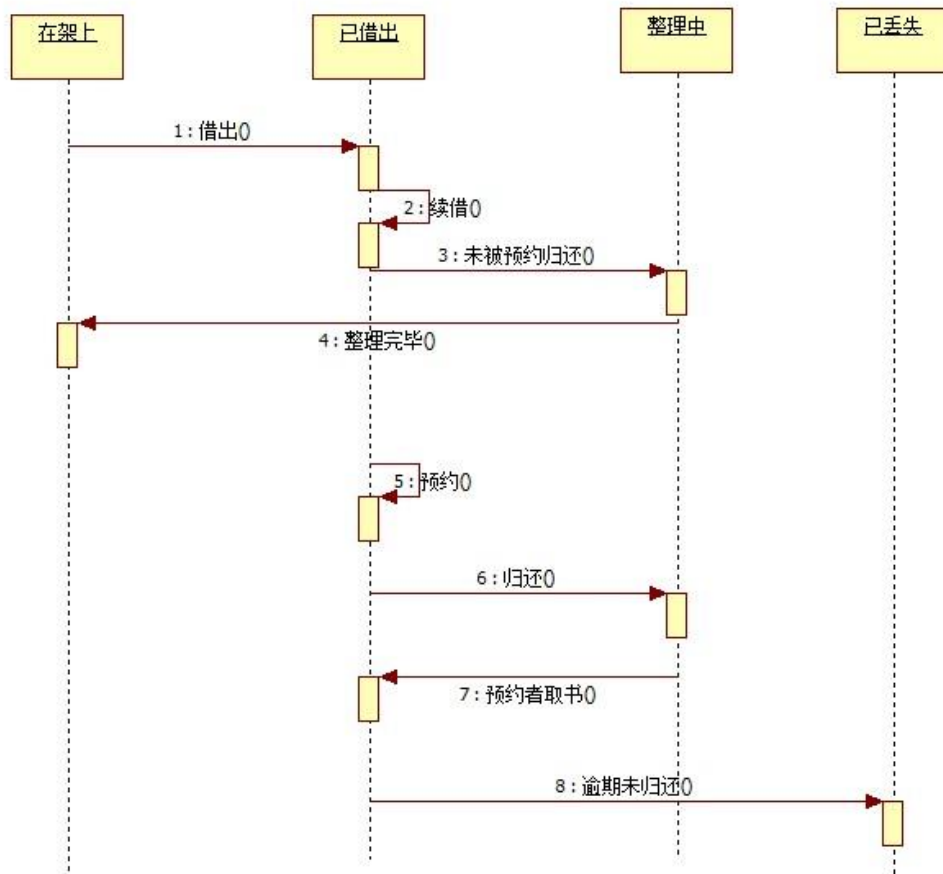
```
__init__.py
forms.py
initmodel.py
models.py
urls.py
views.py
templates
static
fixture
```

2.3 系统类结构图、序列图与网页设计关系图

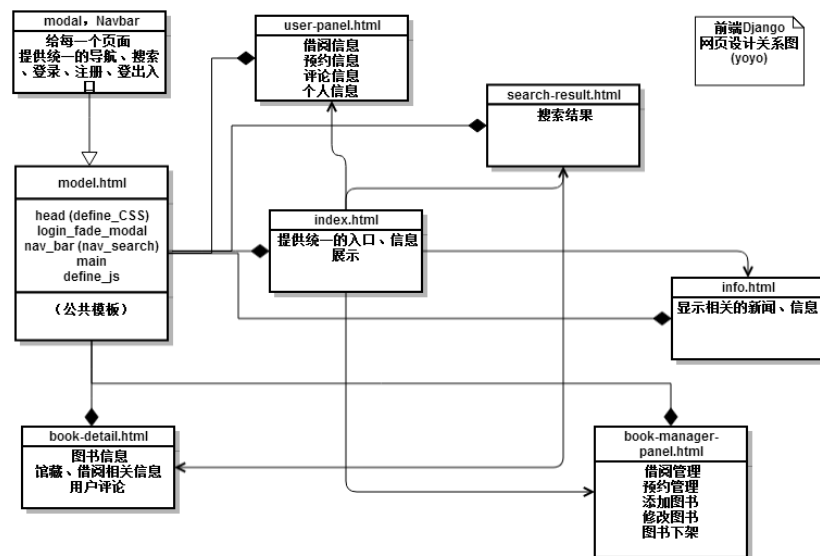
类结构图：



序列图:

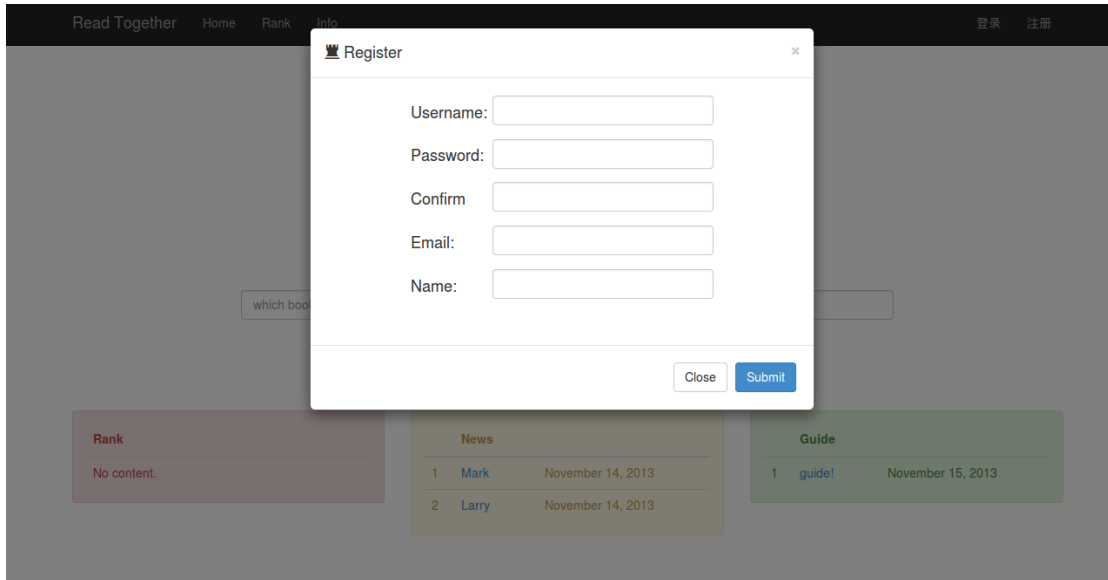


网页设计关系图:



2.4 主要界面流程描述

2.4.1 注册界面流程



The screenshot shows a 'Register' modal window overlaid on the ReadTogether website. The modal contains the following fields and buttons:

- Username:
- Password:
- Confirm:
- Email:
- Name:
- Buttons: Close, Submit

The background website shows a navigation bar with 'Read Together', 'Home', 'Rank', and 'Info'. On the right, there are links for '登录' (Login) and '注册' (Register). The main content area includes a 'Rank' section with 'No content.', a 'News' section with two items, and a 'Guide' section with one item.

News	
1	Mark November 14, 2013
2	Larry November 14, 2013

Guide	
1	guide! November 15, 2013

点击注册按钮输入信息。

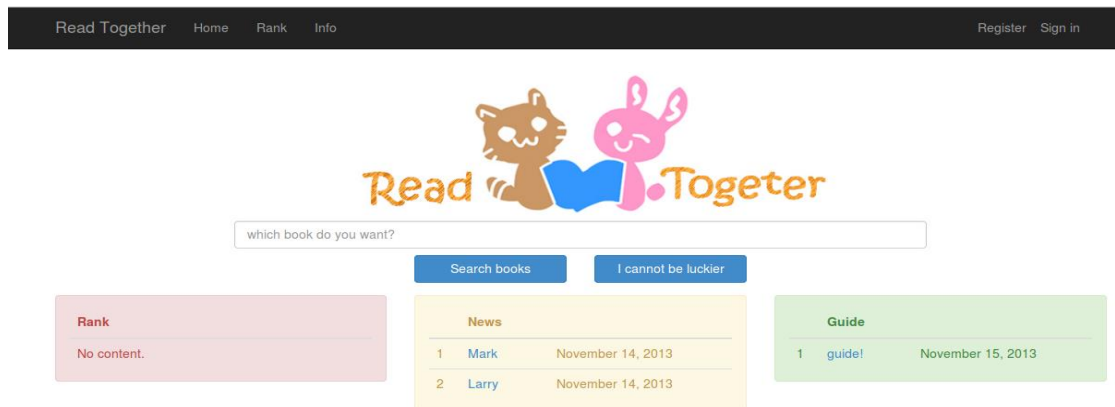


This screenshot shows the website header after a successful registration. The navigation bar now displays 'Read Together', 'Home', 'Rank', and 'Info'. On the right, the links are 'zhangsihao' and '登出' (Logout).



注册成功即可在右上角查看到信息。

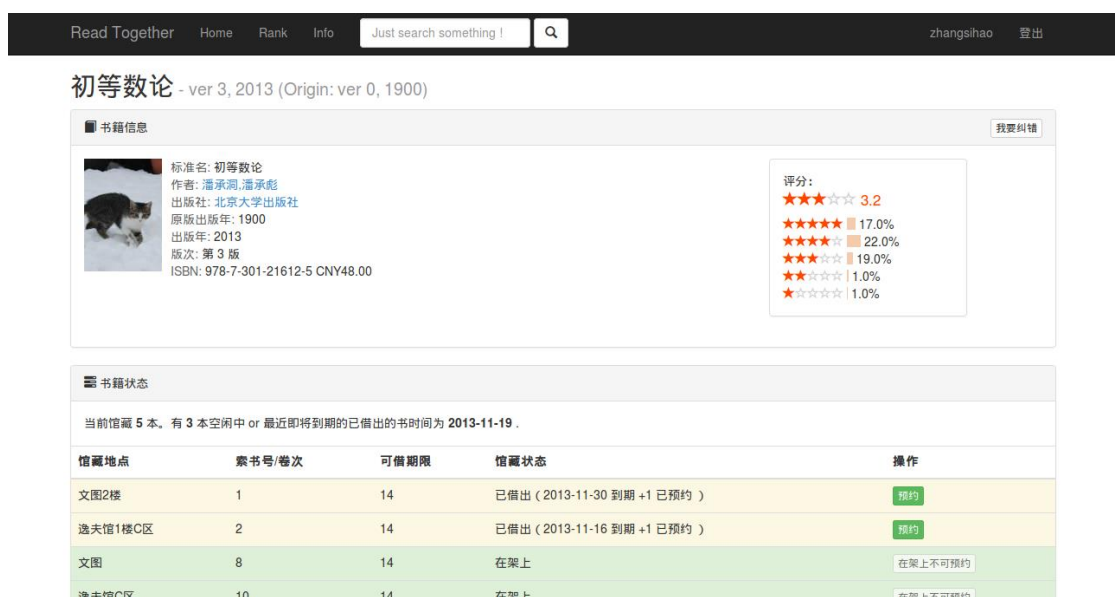
2.4.2 借书界面流程



在搜索框中输入需要查找的图书信息。

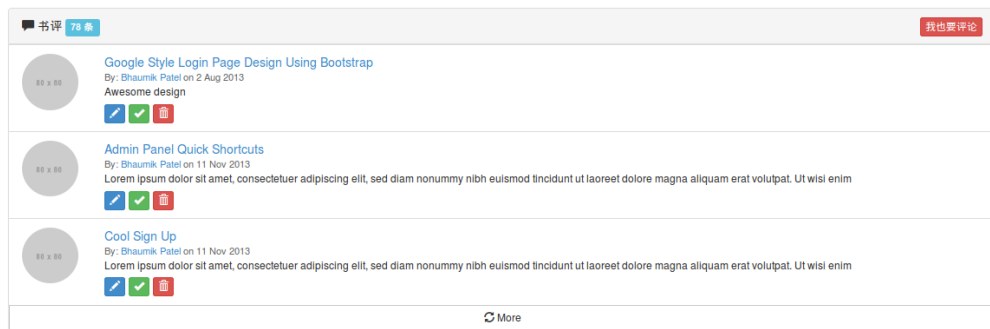


返回搜索结果。

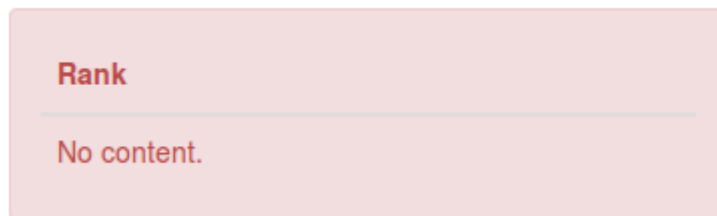


得到图书信息页，提示剩余数量，可在下方选择预约服务。

2.4.3 榜单与评论功能



图书信息页中的评论界面



主页中每月更新图书榜单，尚未正式上线故内容暂时为空。

2.5 模块列表

模块名称(英文)	功能	备注
图书查询模块 Search Module	对所有用户提供图书查询功能	
用户模块 User Module	管理用户信息与行为	
借阅行为模块 Circulation Module	执行不同的借阅行为	
评论模块 Judgment Module	实现用户对图书的评论	
后台管理模块 Management Module	为管理员帐户提供必要的功能	
榜单模块 Ranking Module	统计数据并定期得出图书榜单	
安全、性能模块 Safety Module	提升系统安全性与性能	

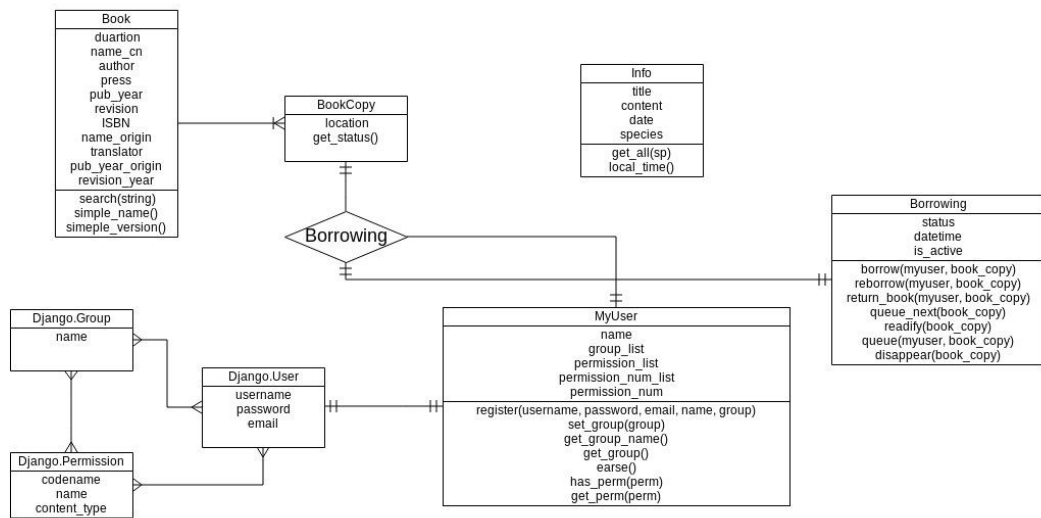
3 数据设计

3.1 数据库表结构:

类	域	对应类	备注
BookCopy	bookcopy	book	表示每个图书可以在不同的地方有多本
Myuser	myuser	user	表示每个用户的用户名都有对应的真实姓名
Borrowing	book_copy/myuser	BookCopy/MyUser	表示一个用户可以同时发生复数次借阅行为,同时借阅多本书

注: 每一行分别代表该类在某一个域中对应的类, 以及详细说明。

3.2 数据库 ER 图:



4 接口规范

4.1 图书模块

描述:

提供图书的管理、查询与相关信息、状态显示功能。

4.1.1 接口: Book

接口函数:

- `simple_name()`: 返回简单书名
- `simple_version()`: 返回简单版本信息
- `search(string)`: static method, 对于馆藏进行以中文书名, 原书名, 作者名进行搜索, 返回搜索结果的交集

4.1.2 接口: BookCopy

接口函数:

- `get_status()`: 返回书籍状态, 其中状态对应编码为 `((0, 'ready'), (1, 'unreturned'), (2, 'over time'), (3, 'arranging'), (4, 'off shelf'))`

4.2 用户模块

描述:

提供用户信息与状态的管理与显示。

4.2.1 接口: MyUser

接口函数:

- `register(self, username, password, email, name, group='NormalUser')`: 用户注册, 需要填写用户名 (会检查是否已存在), 邮箱 (格式有检查), 密码, 确认密码 (有检查), 真实姓名。
- `set_group(self, group)`: 设定用户所拥有的权限。
- `get_group_name(self)`: 供用户查询自己当前权限水平。
- `get_group(self)`: 获得用户的权限的 query set。
- `erase(self)`: 删除该用户。
- `has_perm(self, perm)`: 用于查询布尔权限, 了解该用户是否拥有当前权限 (如是否可进行预约等)。
- `get_perm(self, perm)`: 用于查询整数权限, 了解该用户的该权限的具体数字 (如可借书数量, 借书日期等)。

4.3 借阅行为模块

描述:

提供用户借还、预约、续借图书的管理。

4.3.1 接口: Borrowing

接口函数:

- `borrow(myuser, book_copy)`: 表示该用户借阅了某本特定图书
- `reborrow(myuser, book_copy)`: 表示该用户续借了某本特定图书

- `return_book(myuser, book_copy)`: 表示该用户归还了某本特定图书
- `queue_next(book_copy)`: 在某本书归还后, 将其交给第一个预约的人
- `readify(book_copy)`: 表示某本书归还后已完成整理, 重新上架
- `queue(myuser, book_copy)`: 用户预约某本图书
- `disappear(myuser, book_copy)`: 用户已将该本书遗失

4.4 评论模块

描述:

提供用户在书籍信息页对书籍的评论与打分功能。

4.4.1 接口: comment

接口函数:

- `likeit(sp=None)`: 点赞数加 1。

4.5 后台管理模块

描述:

以管理员为核心的图书馆管理与信息发布系统。

4.5.1 接口: info

接口函数:

- `local_time()`: 以类的形式返回本地时间。
- `get_all(sp=None)`: 返回 `sp` 为某种类的所有 `Info`, `sp=None` 时返回所有 `Info`。
- `get_species_display(str)`: 即上一条的对应关系。

4.6 榜单模块

描述:

在首页中向用户提供每月更新的图书榜单。

无接口

5 模块设计

5.1 图书模块设计

模块设计描述

1) 在 `model.py` 文件中，涉及到图书的属性包括 `book` 和 `bookcopy` 两个类，而在 `Book` 类中，我们设计了一个包括图书的所有属性的表，其中包括以下域：

- `duration`: 借书期限，以整数形式保存，可能取值为 0, 7, 14
- `name_cn`: 图书中文名，若为中文书、中译本或影印本则直接为书名，若对于外文原本则赋空值。
- `author`: 作者名，中英文名称皆支持。当有多个作者时，以“,”分割
- `press`: 出版社名称，中英文名称皆支持。
- `pub_year`: 出版年份，代表公元纪年 4 位数字，以整数形式保存。
- `revision`: 版次，一个阿拉伯数字，以整数形式保存。
- `ISBN`: 根据国际标准，每本书都有一个对应的 ISBN 码。
- `name_origin`: 中译本，影印本，外文原本的外文书名。
- `translator` 译者名称。：影印本和原本为空值。中译本为译者，若有多个人则以“,”分割
- `pub_year_origin`: 原著出版年份，代表公元纪年 4 位数字，以整数形式保存
- `revision_origin`: 原著版次，原著第 X 版相应的阿拉伯数字，以整数形式保存

在 `BookCopy` 类中我们设计了一个包括对某个特定的某本书的所有属性的表，其中包括以下域：

- `status`: 状态，为整数值，对应关系为 `status_choice = ((0, 'ready'), (1, 'unreturned'), (2, 'over time'), (3, 'arranging'), (4, 'off shelf'))`。
- `get_status_display`: 状态，为字符串，即上一条的对应关系。
- `book`: `ForeignKey` 连接其对应的 `book` 类。
- `reborrow_time`: 可以续借的次数。
- `location`: 本书的馆藏位置。

2) 在 `urls.py` 文件中，通过正则表达式匹配到与图书有关的页面的链接，包括图书搜索页面，该书对应的信息页面等

3) 在 `views.py` 文件中，`controller` 实现前段与后端的交互，使得查询的请求与结果得到传送。

4) 在前端文件中，图书信息在 `index.html`, `searchResult.html`, `book-detail.html` 中，分别显示主页，图书搜索结果，图书信息。

5.2 用户模块设计

模块设计描述

1) 在 `model.py` 文件中, 涉及到用户的属性为 `MyUser` 类, 其具体实现如下:

- `User`: 一个对象, 用来表示用户名
- `Name`: 该用户的真实姓名
- `group_list`: 用户当前的权限级别, 包括普通用户, 高级用户, 黑名单, 管理员等。
- `permission_list`: 表示用户所拥有的各种权限, 如“可以借阅, 可以评论”
- `permission_num_list`: 表示某一特定级别的用户的明显权限的具体数字, 包括可借书数量, 借书时间系数, 预约图书数量
- `permission_num`: 对于不同等级的用户, 对他们的 `permission_num_list` 赋具体值

2) 在 `urls.py` 文件中, 通过正则表达式匹配到用户个人信息, 以及注册, 登入登出对应的页面信息所在链接。

3) 在 `views.py` 文件中, `controller` 实现前段与后端的交互, 使得需要调用用户有关信息的请求与结果得到传送。

4) 在前端文件中, 用户的登录和注册显示在首页右上角, 登陆后右上角提供登出按钮和进入个人主页的页面。

5.3 借阅行为模块设计

模块设计描述

1) 在 `model.py` 文件中, 涉及到借阅行为的属性为 `Borrowing` 类, 其具体实现如下:

域

- `status_choice`: 一本书可能的状态, 包括在架上, 已借出, 整理中等
- `status`: 某一本书的特定状态
- `datetime`: 可借阅时间
- `book_copy`: 表示被借阅的某一图书的特定的一本
- `myuser`: 表示该借阅行为中的借阅者
- `is_active`: 表示该图书当前是否已过期

2) 在 `urls.py` 文件中, 通过正则表达式匹配到图书的借阅状态, 以及用户当前借阅。预约的图书对应的页面信息所在链接。

3) 在 `views.py` 文件中, `controller` 实现前段与后端的交互, 使得需要调用借阅行为有关信息的请求与结果得到传送。

4) 在前端文件中, 一本图书的信息页面上会显示它的所有 `copy` 的借阅状态, 在用户页面会显示自己所有借阅及预约行为的各项状态。

5.4 评论模块设计

模块设计描述

1) 在 `models.py` 文件中, 涉及到首页信息的属性为 `comment` 类, 其具体实现

如下:

- content 内容
 - date 时间 (自动存入创立时的时间)
 - user 用户 (发送人)
 - book 所评论的书
 - zan 被点赞的数目
- 2) 在 views.py 文件中, controller 实现前段与后端的交互, 使得调用评论的请求与结果得到传送。
- 3) 在前端文件中, 评论信息显示在指定的图书信息页, 具体内容显示在 bookdetail.html 模板中。

5.5 后台管理模块设计

模块设计描述

- 1) 在 model.py 文件中, 涉及到首页信息的属性为 info 类, 其具体实现如下:
- title 标题
 - content 内容
 - date 时间 (自动存入创立时的时间)
 - species 种类, 整数域, 存入 `species_choice=((0, 'news'),(1, guide))`
- 2) 在 urls.py 文件中, 通过正则表达式匹配到公告对应的页面信息所在链接。
- 3) 在 views.py 文件中, controller 实现前段与后端的交互, 使得调用公告的请求与结果得到传送。
- 4) 在前端文件中, 图书公告信息的标题显示在首页, 具体内容显示在 info.html 中。

5.6 榜单模块设计

模块设计描述

- 1) 在 model.py 文件中, 涉及到榜单信息的属性为 board 类, 其具体实现如下:
- title 书名
 - position 名次
- 2) 在 urls.py 文件中, 通过正则表达式匹配到公告对应的书籍信息所在链接。
- 3) 在 views.py 文件中, controller 实现前段与后端的交互, 使得调用公告的请求与结果得到传送。
- 4) 在前端文件中, 榜单信息的标题显示在首页, 具体内容显示在 info.html 中。

6 附录

6.1 参考资料

Python 官方文档
Django 官方文档
MySQL 官方文档
BootStrap 文档
JQuery 文档
CSS 文档
HTML5 标准