

# Novels-Searching and Reading Web Application

*quick reading*

Semester 2 Report

Yan Liu

20082245

Supervisor: Anita Kealy

BSc (Hons) in Software System Practice Year 4

## Table of content

<b>1.0 ABSTRACT .....</b>	<b>4</b>
<b>2.0 INVESTIGATION .....</b>	<b>4</b>
2.1 BUSINESS OPPORTUNITY .....	4
2.1 MARKET ANALYSIS .....	5
2.1.1 <i>Market Situation</i> .....	5
2.1.2 <i>User Analysis</i> .....	5
2.2 COMPETITOR ANALYSIS .....	5
2.2.1 <i>The competition within the other novel reading website</i> .....	5
2.2.2 <i>The bargaining power of customers and suppliers</i> .....	6
<b>3.0 APPLICATION REQUIREMENTS.....</b>	<b>6</b>
3.1 FUNCTION REQUIREMENT.....	6
3.1.1 <i>Web crawlers function</i> .....	6
3.1.1.1 <i>Get search result data from search engine</i> .....	6
3.1.2 <i>Searching function</i> .....	7
3.1.3 <i>Novels ranking function</i> .....	7
3.1.4 <i>Register function</i> .....	7
3.1.5 <i>Login function</i> .....	8
3.1.6 <i>Reading function</i> .....	8
3.1.7 <i>Bookshelf function</i> .....	9
3.1.8 <i>Bookmarks function</i> .....	10
3.1.9 <i>Email notification function</i> .....	10
3.1.10 <i>Payment function</i> .....	11
3.1.11 <i>Comment function</i> .....	11
3.2 NON-FUNCTION REQUIREMENT.....	11
3.2.1 <i>Performance requirements</i> .....	11
3.2.2 <i>Security requirements</i> .....	11
3.2.3 <i>Reliability requirement</i> .....	11
3.2.4 <i>Compatibility requirements</i> .....	12
<b>4.0 METHODOLOGY .....</b>	<b>12</b>
4.1 AGILE SOFTWARE DEVELOPMENT .....	12
<b>5.0 TECHNOLOGIES AND TOOLS.....</b>	<b>12</b>
5.1 DEVELOP LANGUAGE .....	12
5.2 PYTHON AND PACKAGES VERSION CONTROL.....	12
5.3 CODE VERSION CONTROL.....	13
5.4 PYTHON WEB CRAWLER.....	13
5.5 CACHE TECHNOLOGY .....	13
5.6 EMAIL NOTIFICATION TECHNOLOGY .....	13
5.7 PAYMENT TECHNOLOGY .....	14
5.8 ROLE-BASED ACCESS CONTROL TECHNOLOGY .....	14

5.9 SERVER TECHNOLOGY .....	15
5.9.1 <i>Sanic framework</i> .....	15
5.9.2 <i>Blueprint</i> .....	15
5.9.3 <i>Server configuration</i> .....	15
5.9.4 <i>Gunicorn configuration</i> .....	15
5.10 DATABASE TECHNOLOGY .....	15
5.10.1 <i>MongoDB</i> .....	15
5.10.2 <i>Redis</i> .....	16
5.11 CLIENT.....	16
5.11.1 <i>Bootstrap framework</i> .....	16
5.11.2 <i>Jinja template</i> .....	16
5.11.3 <i>jQuery</i> .....	16
5.11.4 <i>Ajax</i> .....	16
5.12 TESTING .....	16
5.13 CODE QUALITY.....	17
5.14 DEPLOYMENT.....	17
<b>6.0     DIAGRAMS.....</b>	<b>17</b>
6.1 USE CASES DIAGRAM .....	17
6.2 CLASS DIAGRAM .....	19
6.3 ACTIVITY DIAGRAM .....	20
6.4 TECHNICAL DIAGRAM .....	21
6.4 INTERFACE DESIGN DIAGRAM .....	22
<b>7.0     DEVELOPMENT SCHEDULE .....</b>	<b>27</b>
7.1 GANTT CHART.....	27
<b>8.0 PROBLEMS ENCOUNTERED AND RESOLVED.....</b>	<b>30</b>
8.1 PYTHON ENVIRONMENTAL MANAGEMENT .....	30
8.2 HOW TO USE OF DATABASES IN PYTHON PROJECT .....	30
8.3 HOW TO GET NOVELS DATA FROM INTERNET .....	30
8.4 PAYMENT SYSTEM.....	30
8.5 HOW SEND HTML EMAIL.....	31
8.6 DEPLOY PROBLEM .....	31
<b>9.0 SELF-REVIEW .....</b>	<b>32</b>
9.1 WHAT I LEARNT .....	32
9.2 ADDITIONAL FEATURES TO ADD .....	33
<b>10.0 REFERENCE .....</b>	<b>34</b>

# **1.0 Abstract**

With the popularity of the Internet, the way books exist has changed, more and more e-books have appeared. And the development of mobile phones makes it easier for people to browse websites. Especially in China, the novel has entered the "site era", that is, relying on large-scale literary websites, such as Qidian Chinese network. (Young 、K, 2018) According to Sina weibo, China has 333 million users and a market of 9 billion yuan, with an average annual growth rate of 20%. Reading novels through the website is already a trend, which allows readers to easily arrange their reading time, such as waiting for the car, you can read it. The purpose of this project is to make it easier for readers to read novels and provide better service for them. Users can register their own accounts, collect favorite books, save their own reading records, and be free from other Spam advertising, and get updated reminders for books which they like. Therefore it is a good place for reader to enjoy the interest of reading.

This project focuses on searching novels and reading experience, mainly focusing on Chinese novels. It has a search engine which is based on other search engines (Baidu, 360, Bing, Google). It doesn't store any novel data (but it will cache novels which are searched in order to make it faster to get data next time), all the data is discovered by crawler technology based on third-party novel websites, and then displayed to users through online analysis. It is multi-website aggregation and has high selectivity. It doesn't have any advertising because the advertising and any script will be deleted when parsed. The project uses MongoDB to store basic information generated during users' use, such as registration information, search records, reading history data, bookshelf and bookmark data, etc. For some necessary caches, such as novel cache and session cache, Redis is used for cache processing.

# **2.0 Investigation**

## **2.1 Business Opportunity**

According to the Amazon China 2015 National Reading Survey Report, which was conducted by Amazon China and Xinhua website, collected data from 14,000 readers in more than 500 cities across the country through multiple channels. The report shows that there is a strong atmosphere for people to read, and reading has become almost a necessity for most of the respondents. Most of the respondents read novels almost every day, and 36% read more than one hour a day. Nearly half of the respondents maintained the habit of reading a large number of novels. Over the past year, they read more than 10 books, almost one book a month. The novel has many characteristics: rich content, changeable plot and attractive to readers. In addition, the length of the novel is long,

which can attract loyal fans to pursue books and increase user traffic for the novel website. Therefore, the market for novel reading websites is very broad and still in the stage of development, which has great commercial value.

## **2.1 Market Analysis**

### **2.1.1 Market Situation**

The diversified communication mode of network media is conducive to the spread of novels and brings a large number of users to novel reading websites. In addition, the convenience of mobile phones and 4G networks makes it easier for readers to read novels through the Internet, and the popularity of mobile phones has expanded the audience of novels. In 2017, the mobile reading market in China reached 15.318 billion yuan, 29.16% higher than last year, due to a sharp increase in revenue from copyright and paid reading. In the future, the concept of paid reading for users will be further mature. Literature, film and television, animation, games and other industries will integrate with each other.

### **2.1.2 User Analysis**

Actually, there are lots of difference in reading habits between the Chinese and the Irish. According to survey, in china, the scale of mobile reading users has been growing steadily: in 2017, the scale of online literature users has reached 378 million, with a year-on-year growth of 13.37%. The number of novel readers is mainly young people. In the process of the development of novel reading websites, the number of female users has been increasing. Students and office workers account for the largest proportion, and young users have a stronger sense of payment. But in Ireland, people prefer literary books, I often can see them reading books (especially paper books) in the bus or train or plane. When I asked Irish schoolmates, they tell me they would like play games than reading novels. So, this website mainly focus on Chinese novels, and think about copyright, I would like to choose chinses novels website to get novels data. Therefore, when I design this website, more consideration is given to Chinese aesthetics and senses.

## **2.2 Competitor Analysis**

### **2.2.1 The competition within the other novel reading website**

First, the industry growth is slow and the competition for market share is fierce. Second, the number of competitors is large, the competitive power is generally equal. Third, the

products or services provided by competitors are roughly the same, or at least do not show obvious differences. Fourth, in order to benefit from economies of scale, some enterprises expand their production scale, break the market balance, produce a large number of surplus products, and begin to resort to cut-price competitive sales.

## **2.2.2 The bargaining power of customers and suppliers**

Customers are the users of website. The bargaining power of customers lies in whether they can urge the seller (who provide novels) to lower the price, improve the product quality or provide better service. Bargaining power of suppliers refers to whether suppliers can effectively urge buyers to accept higher prices, earlier payment time or more reliable payment methods.

# **3.0 Application Requirements**

## **3.1 Function Requirement**

### **3.1.1 Web crawlers function**

#### **3.1.1.1 Get search result data from search engine**

Use BeautifulSoup and urllib tool to get the search result of Baidu, 360 So, Bing, Google search engine.

#### **3.1.1.2 Get novels chapter data from other novels website**

Use BeautifulSoup and urllib tool to get the chapter data which contains url and chapter name from other novels website.

#### **3.1.1.3 Get novels content data from other novels website**

Use BeautifulSoup and urllib tool to get the content of selected novels' chapter from other novels website.

## **3.1.2 Searching function**

### **3.1.2.1 Search by novel name**

User input the name of novel in the search bar, then turn to the search result page.

### **3.1.2.2 Search by author name**

User input the name of author in the search bar, then get all of the novels which written by the author in the search result page.

### **3.1.2.3 Search record**

It will store novels name and search counts (which can update when search again) in MongoDB. It also store search record of a user.

## **3.1.3 Novels ranking function**

The twenty five most searched novels names are arranged in the form of a label below the search bar, when user click on it, it will automatic search that novels and turn to the search result.

### **3.1.3.1 Novels Ranking list**

Use search record(which is stored in MongoDB), show the novels search counts in form of table. User can sort it by novel name or search count. Click the novel's name, it will automatic search that novels and turn to the search result.

## **3.1.4 Register function**

Users register their own account. The register button and login button are in the same dialog box. When users click on the Login button in the upper right corner, they enter the dialog box which contains register button. Click on the Register button, enter the register page. User need to input user name, email, verification code, password and repeat password to register a account. The length of user name need to greater than or equal to 2 bits while password length is greater than or equal to 6 digits. Email need to have right format and the repeat password needs to be right. Otherwise, error message will be prompted in register page.

### **3.1.5 Login function**

#### **3.1.5.1 General user login**

User login their account and reload the page, user drop down list will replace the login button on the top right corner. Because it use session to store the user account information. User login in and cookies expire date will add 30 days and that means user do not need to login again next time. If use logout, he need to login again.

#### **3.1.5.2 VIP user login**

VIP user login in and have different drop down list.

#### **3.1.5.3 Administrator login**

Administrator login and have different drop down list.

### **3.1.6 Reading function**

#### **3.1.6.1 Searching result page**

The searching result page is based on 360 or Baidu or Bing or Google search result. By default, it select 360 search results to generate search results page in this website(quickreading). The searching result page contains the url of chapter page in other website which is use to get chapter data to build quickreading's chapter page. It also contains the url of quickreading's chapter page and the novel's title.

#### **3.1.6.2 Chapter page**

If user click one url in search result, for the site(url) has been parsed, it will enter the quickreading's chapter page, which is developed by me, contains the chapters of novel, when user click on one chapter, it will enter content reading page. And for the unparsed website url, it will jump to the chapter page in original site.

### **3.1.6.3 Content page**

Users read novels in this page. The color of background in the page may be light yellow, which protect readers' eye. The default of word size is 14, font is Times New Roman. It also contains search bar in the upper left corner and user action bar in left section which contains chapter button to help user go to the chapter page, bookshelf button to help user go to their user interface which contains novels they like, home button to help user go back index page.

### **3.1.6.4 Adjust the font**

User can adjust the format of words including the size, font.

### **3.1.6.5 Adjust the background**

User can adjust the background color of reading page. They can choose six themes they like provided by the website.

## **3.1.7 Bookshelf function**

### **3.1.7.2 Add novel in bookshelf**

User who registered account and login can add the reading page into the bookshelf of him by click on the bookshelf button in left section. If user not login, he can not add it, and will get site alerts to login.

### **3.1.7.3 Access bookshelf**

Only VIP user can access to the bookshelf. Novels information such as novel name, chapter url, latest chapter url, reading history are shown in bookshelf.

### **3.1.7.4 Delete novel in bookshelf**

VIP user can delete the records in his bookshelf by click Delete button.

### **3.1.8 Bookmarks function**

#### **3.1.8.1 Add novels in bookmark**

User who registered account and login can add the reading page into the bookshelf of him by click on the bookmarker-shaped buttons. If user not login, he can't add it, and will get site alerts to login.

#### **3.1.8.2 Access bookmark**

Only VIP user can access to the bookmark. Novels information such as novel name, chapter's url, chapter name, date are shown in bookmark.

#### **3.1.8.3 Delete records in bookmark**

VIP user can delete the records in his bookshelf by click Delete button.

### **3.1.9 Email notification function**

#### **3.1.9.1 Register notification**

When users register successfully, the website will send the register successfully notification email automatically to the email address which user register.

#### **3.1.9.2 Payment notification**

When users payed the vip service, the website will send the payment successfully notification email automatically to the users' email. And the payment notification email is different from according to users' role.

#### **3.1.9.3 Renew notification**

The website will send renew notification email when user's vip service will expire. It will send email the day before the expiration date.

### **3.1.10 Payment function**

User can choose three types of VIP service: one month VIP service, six months VIP service, one year VIP service. Then click on the PayPal button to pay the fee.

### **3.1.11 Comment function**

User can comment on the reading page. It support Multiple ways to log in to comment, such as Email, Google account, Facebook, Twitter.

## **3.2 Non-Function Requirement**

### **3.2.1 Performance requirements**

The response time of searching time of general period is less than 1 seconds, and the peak period is less than 3 seconds.

The response time of fuzzy query is less than 2 seconds

The time of login is less than 2 seconds.

### **3.2.2 Security requirements**

After authentication, the user can only access the data within the scope of his permission and can only do the operations within the scope of his permission.

Different users have different identities and authorities, and it is necessary to provide credible authorization management services on the premise that the user's identity is authentic, protect the data from illegal/unauthorized access and tampering, and ensure the confidentiality and integrity of the data.

### **3.2.3 Reliability requirement**

Prompt for input, check data to prevent abnormal data. The system has strong robustness and should be able to handle all kinds of abnormal situations in the process of system operation, such as: human operation error, illegal data input, hardware device failure, etc., the system should be able to handle correctly and avoid appropriately. The system failure rate is up to 1 failure every 1,000 hours.

### **3.2.4 Compatibility requirements**

The system should support MAC, IOS, Android and Windows operating systems;

The system should support MongoDB and Redis database system.

## **4.0 Methodology**

### **4.1 Agile Software Development**

Agile software development is an approach to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer(s)/end user(s). It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages rapid and flexible response to change. This project will uses following cycle to develop website:

## **5.0 Technologies and Tools**

### **5.1 Develop Language**

- Python3.6: use it to develop the server.
- JavaScript: use it to develop the client.

### **5.2 Python and packages version control**

The version has always been a problem when develop project. I use pipenv ( which is similar to NPM ) and anaconda3 to solve this problem. I use anaconda3 to create python3.6 virtual environment (Real Python, n.d.), then use pipenv to control the packages used in project (Pipenv, 2018). Pipenv can automatically creates and manages a virtualenv for your project, as well as adds/removes packages from your Pipfile as you install/uninstall packages. It also generates the ever-important Pipfile.lock, which is used to produce deterministic builds. That is the reason why I choose it to manage my packages for this project.

## 5.3 Code version control

This project use Git to control the project's version. There are lots of git log. Every time the code is updated and confirmed to be bug free, the code is committed and pushed to the GitHub remote repository.

## 5.4 Python web crawler

This project doesn't store novels information, I mean you do not need to set up a database to store all of novels' data. It will get the novels data from internet. When user input the novel's name in search bar. It use aiohttp to send request and get the search result page from the web (360 so, Baidu, Bing search result). Then use BeautifulSoup and urllib package to parse the html resource(what we get above) to find where is the title and chapter url and generate the url in quickreading. When user selected one chapter url, it will parse the chapter page in other website and find chapter name and the url of the content of every chapter. When user select one chapter to read, it will parse this chapter and next chapter and pre chapter content in other website(website that have been crawled).

## 5.5 Cache technology

Given the user experience, they should expect to be able to quickly search a novel, quickly enter the chapter page, and quickly get the chapter content without waiting for parsing. Therefore, I adopt cache technology. By using aiocache which is an asyncio cache manager for redis to cache the search result and chapter and content. When user search the novel next time, quickreading will find the data from cache(redis), that means it saves a lot of time.

## 5.6 Email notification technology

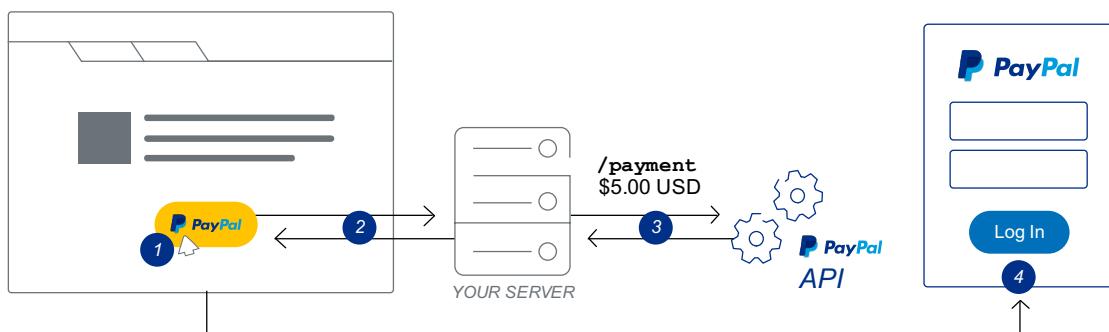
I use Amazon Simple Email Service (Amazon SES) to send email to users. It write code to call the SMTP Interface to send email. Actually, the harder part is how to move your account out of sandbox and getting the credentials and how to send html email according to different user name and email address, that is what I done to build a email system in this project. The process of it shows below:



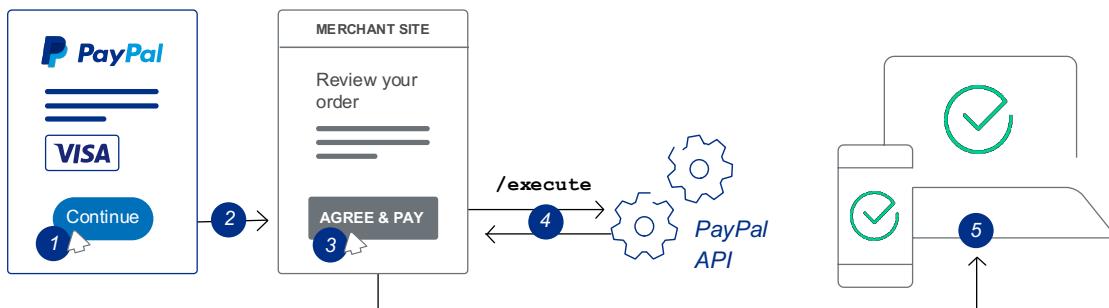
## 5.7 Payment technology

Since it involves payment to become a member, I need to find a way to receive payment. I choose PayPal checkout (Paypal, n.d.), add a smart payment buttons integration to my website client. Then write server routing to deal with the payment call. Overall, this technology has two aspects, one is Set Up a Payment, the other is Execute the Payment. The work flow diagram shown below (Paypal, 2019).

**Set up a Payment:**



**Execute the Payment:**



## 5.8 Role-based access control technology

Because there are three roles in this project. Their permissions have no conflict problem. Therefore, I classify their roles according to the level of authority. The administrator has the highest privileges, including deleting ordinary users and VIP users. Ordinary(general) users can only search for novels, and although they can bookmark novels, they cannot access bookshelf to view them (encouraging people to pay to be VIP).

## **5.9 Server technology**

### **5.9.1 Sanic framework**

Sanic is a Flask-like Python 3.5+ web server which is written to go fast. Sanic supports async request handlers. This project use this framework to build an asynchronous web server.

### **5.9.2 Blueprint**

Blueprints are objects that can be used for sub-routing within an application. Instead of adding routes to the application instance, blueprints define similar methods for adding routes, which are then registered with the application in a flexible and pluggable manner. Blueprints are especially useful for larger applications, where your application logic can be broken down into several groups or areas of responsibility (Sanic, 2018).

### **5.9.3 Server configuration**

Config the website details,such as the host address, timezone, base directory, token.

### **5.9.4 Gunicorn configuration**

Gunicorn is a python WSGI HTTP Server for UNIX. The reason I use it is to prepare for the deployment cause it can set the number of worker processes to spawn, set the bind port.

## **5.10 Database technology**

### **5.10.1 MongoDB**

This project uses MongoDB to store basic information generated during users' use, such as registration information, search records, reading history data, bookshelf and bookmark data, etc. In order to realize asynchronous, I use Motor which is an Asynchronous Python driver for MongoDB.

## **5.10.2 Redis**

This project uses Redis for cache novels data. In order to realize asynchronous, I use asyncio-redis and aiocache to manage the redis database.

## **5.11 Client**

### **5.11.1 Bootstrap framework**

Bootstrap is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile first projects on the web.

### **5.11.2 Jinjia template**

I use Jinjia which is a full featured template engine for Python. It can render pages and reduce the repetition rate of page information (the same elements can be called using `{%extends %}`). In addition, for the data sent by the server, it can make logical judgment and management (for example, using `{%if%}` to control expressions or logic

### **5.11.3 jQuery**

This project use lots of JQuery to handle events, such as button clicks, page swipes, mouse clicks, and so on. The routing of the link server, such as sending Ajax post request to the server, processing the returned data, judgment and so on.

### **5.11.4 Ajax**

This project use Ajax to send request to server(post request).It can refresh a section of a web page instead of re-opening the whole page. It can reduces the burden on the server and improves the smooth running of the client.

## **5.12 Testing**

This project mainly focus on Automated testing which is the act of conducting specific tests via automation as opposed to conducting them manually. I use Selenium to realize automated unit testing. It tests the process of searching and reading novels, user register and login and logout. In addition, I also use Pytest to write some simple test for the client to test the status of response.

## **5.13 Code Quality**

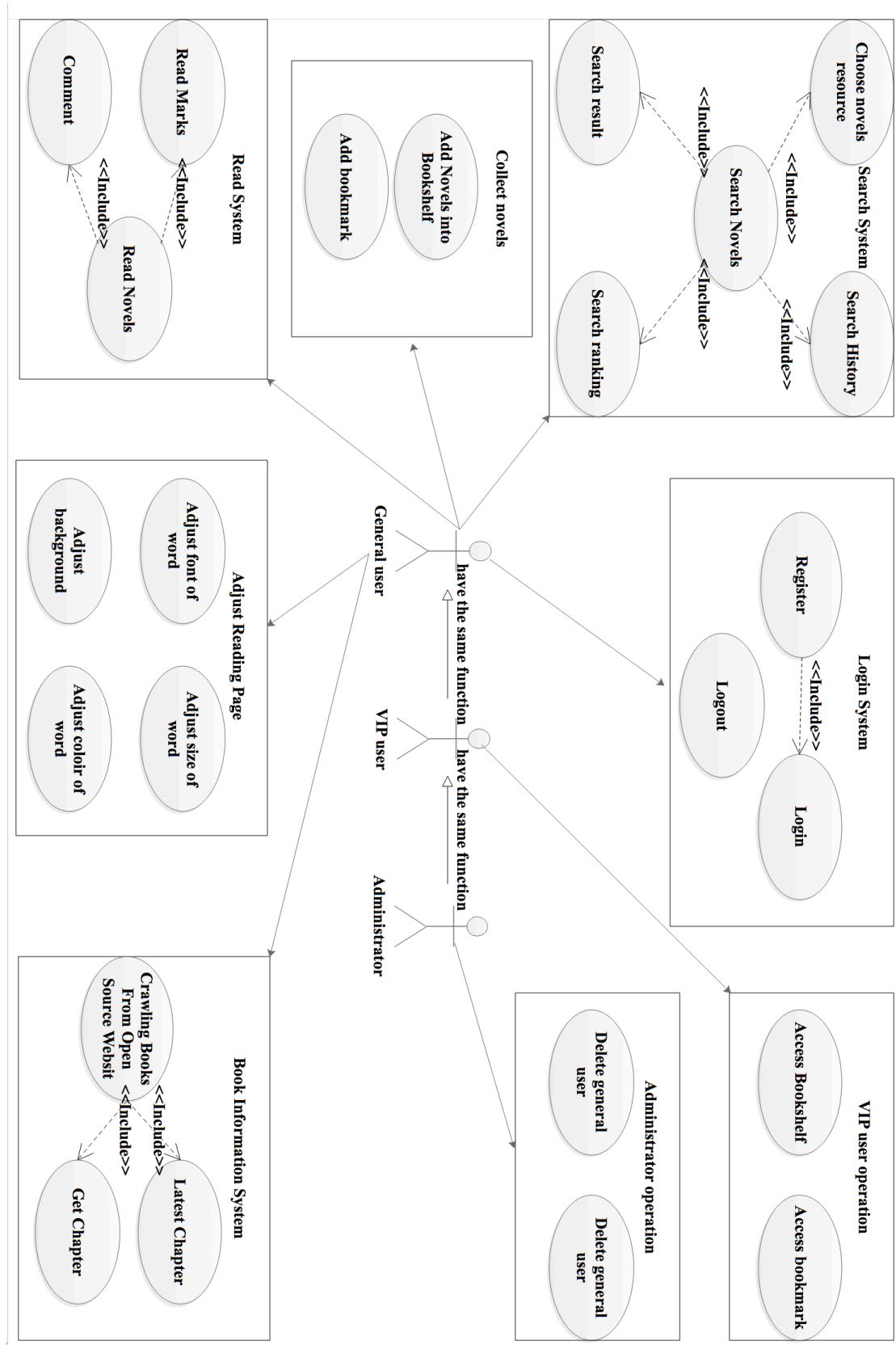
I follow the PEP 8 — the Style Guide for Python Code to write the code. Using PyCharm to develop the project, cause it can remind yourself of code irregularities

## **5.14 Deployment**

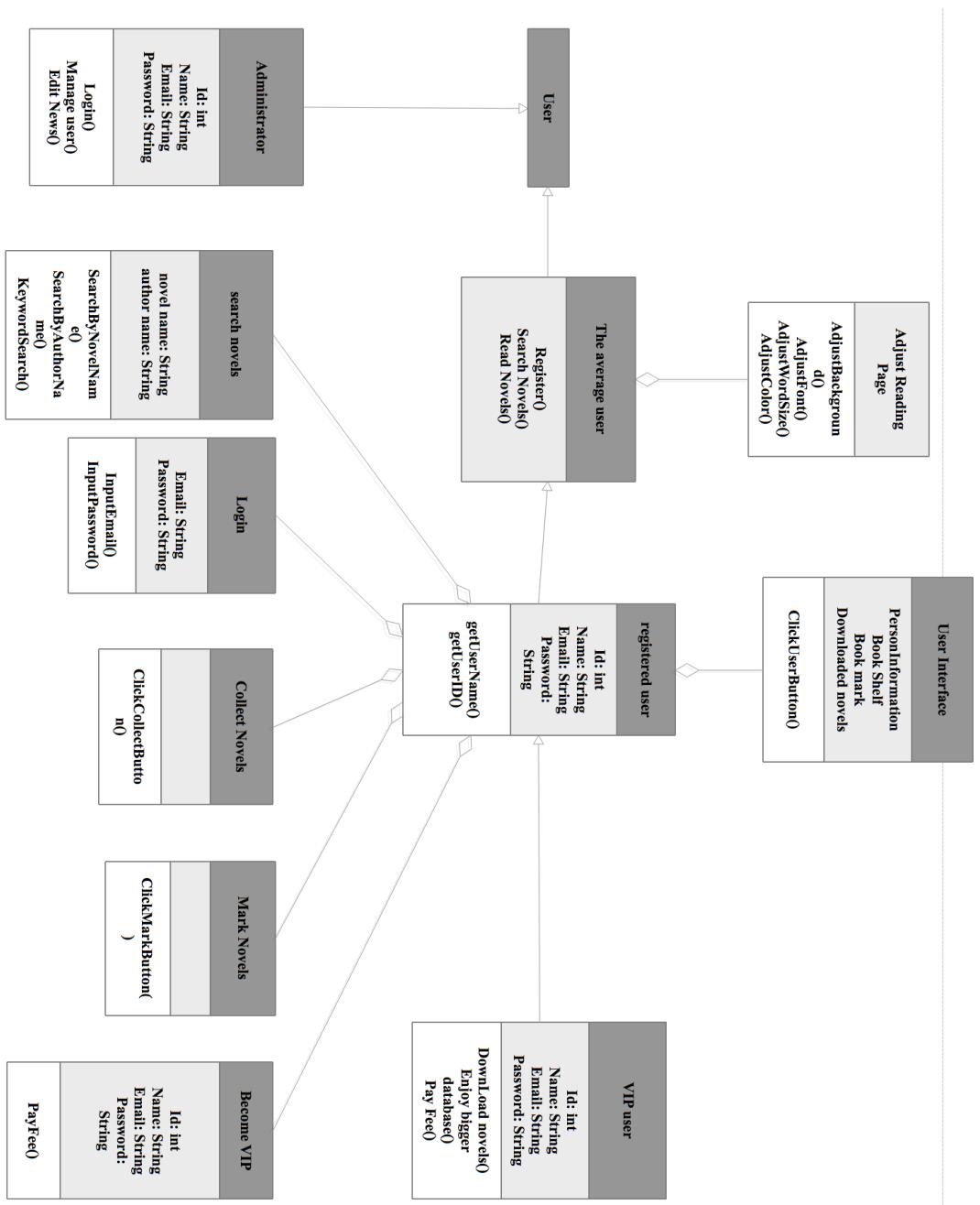
There is no doubt that the ultimate goal of developing a project is to deploy it as a product. In order to learn more about deployment, I choose to try enterprise project deployment approach. First of all, I rent a server on Vultr (IP address is : 45.76.134.147). The server's operating system is CentOS 7x64. Therefore, I learn a lot of Linux operating instructions to configure the server, such as the database, python environment, some tools to install software, etc. Then clone my project from GitHub, and use pipenv to install related packages. Then I bought a domain name called quickreading.net from irishdomains.com and do Domain Name Resolution in aliyun.com. Therefore, the vultr server connect with my domain name. Then run the project in the remote server, and the project is deployed successfully. You can visit it by search: quickreading.net.

# **6.0 Diagrams**

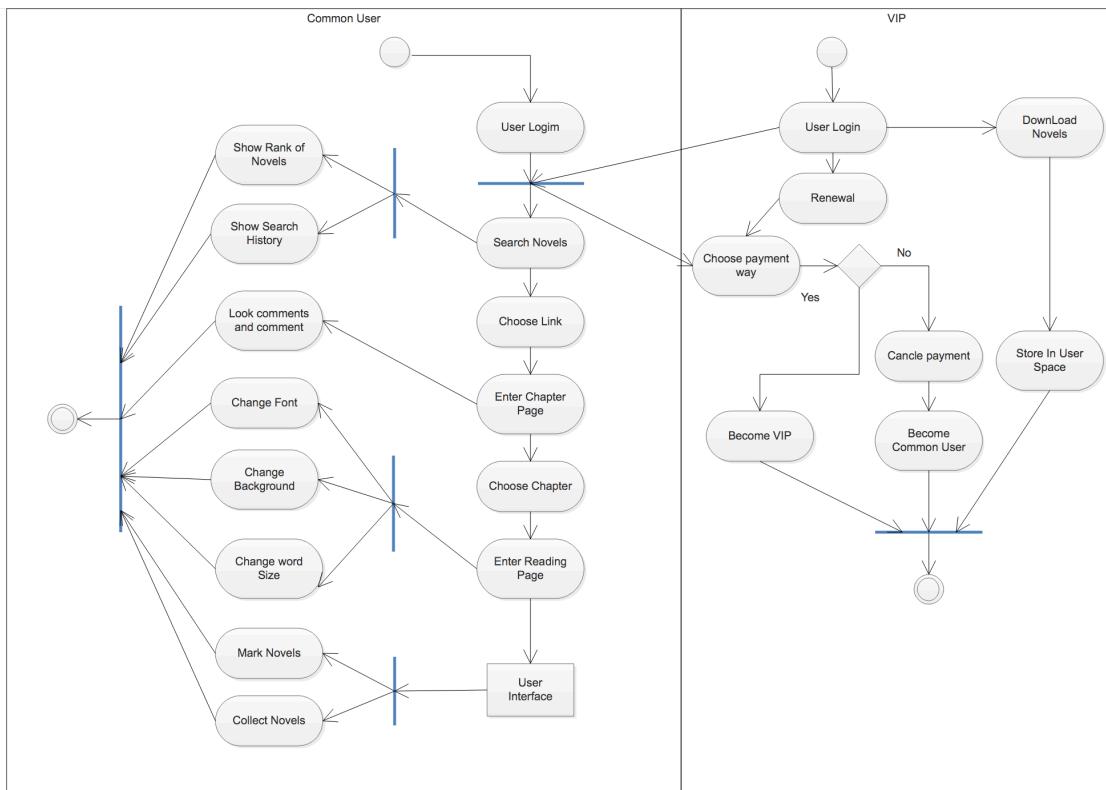
## **6.1 Use cases Diagram**



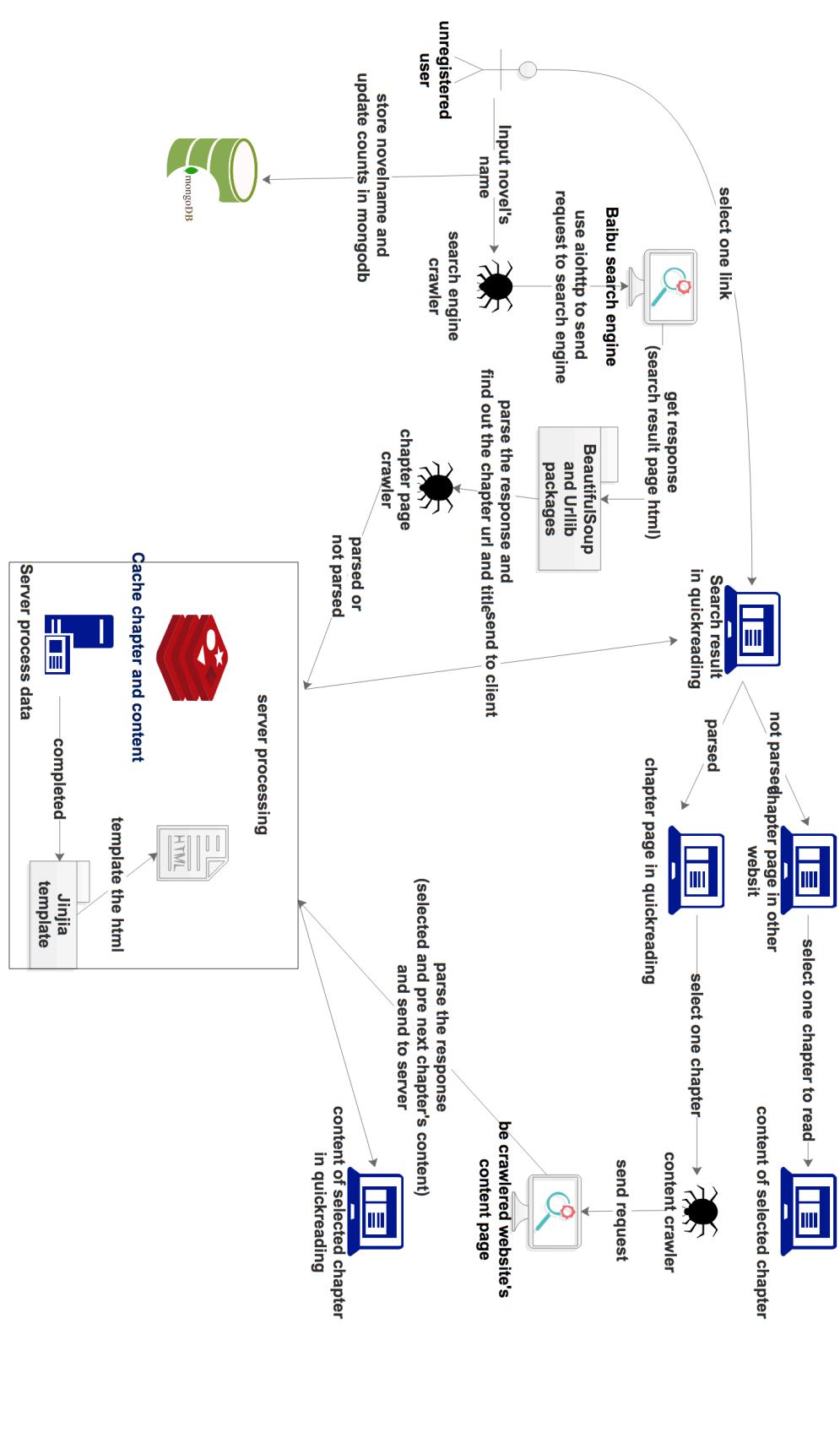
## 6.2 Class Diagram



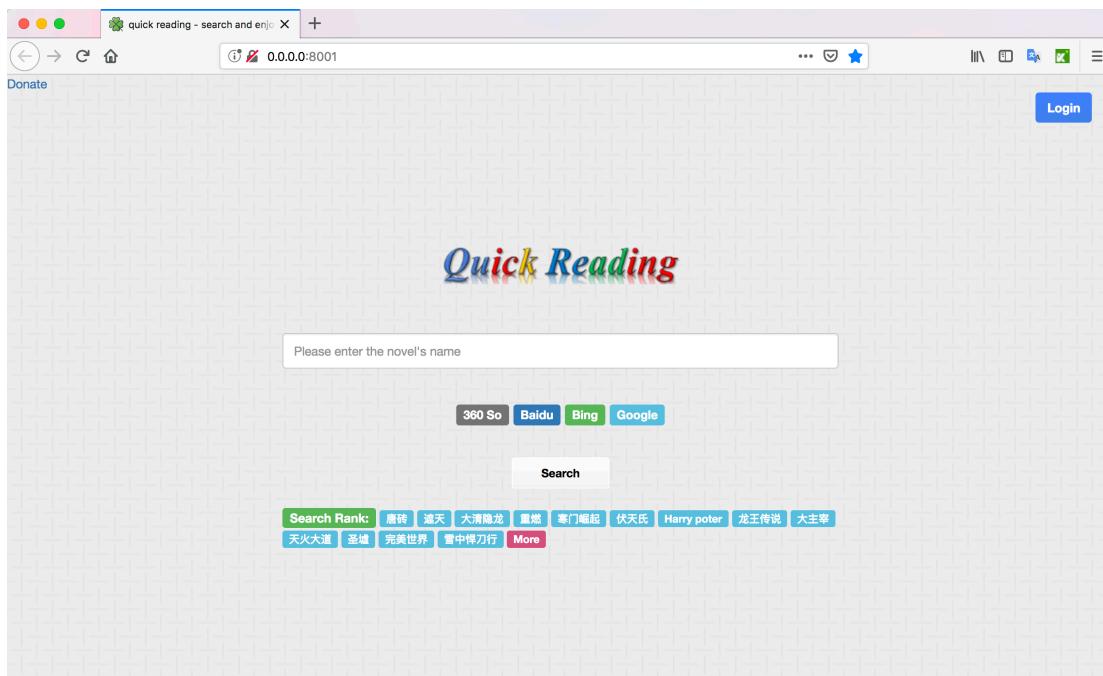
## 6.3 Activity Diagram



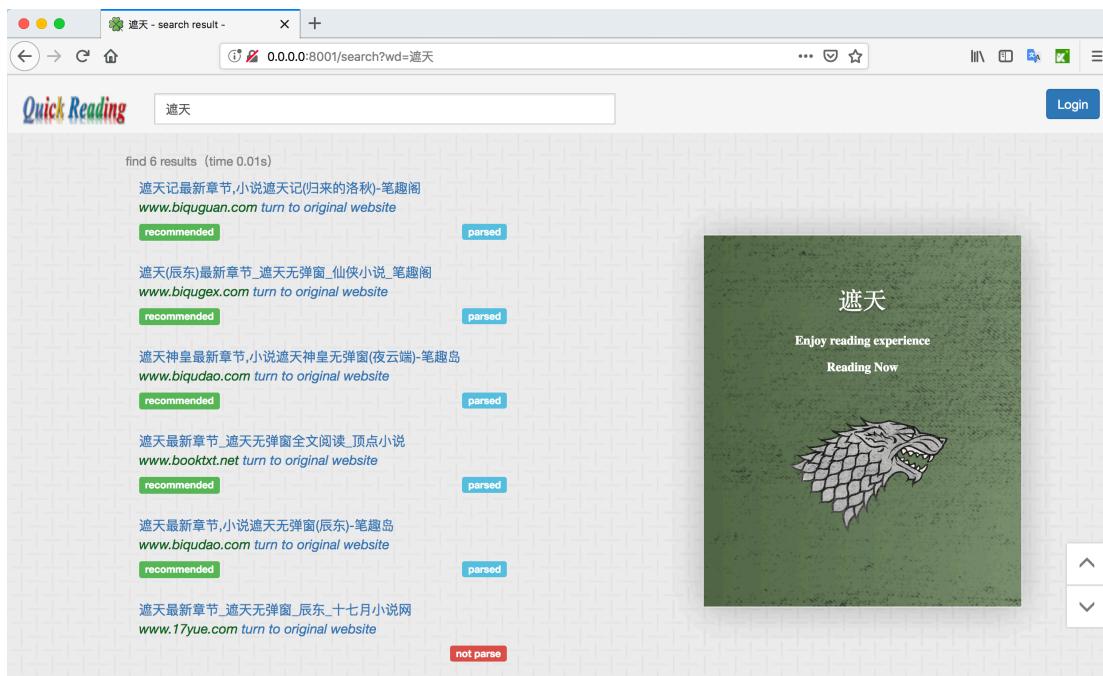
## 6.4 Technical Diagram



## 6.4 Interface Design Diagram



Index page



Search result page

遮天 - quick reading website

我的新书《圣墟》已上传

我的新书《完美世界》已上传,请兄弟姐...

完本感言

《遮天》最新章节

第一千八百二十二章 遮天大结局

第一千八百二十一章 九龙拉棺

第一千八百二十章 时光飞逝

《遮天》正文卷

正文		
第三章 今昔	第一章 星空中的青铜巨棺	第二章 素问
第六章 封禅之地	第四章 荒古铜刻	第五章 李小曼
第九章 铜棺内第三十人	第七章 五色祭坛	第八章 星空古路
第十三章 天宫遗址	第十章 苍茫大地	第十一章 光源
第十六章 如神祇临尘	第十四章 大雷音寺	第十五章 菩提
第十九章 从容	第十七章 生死	第十八章 对立
第二十二章 铜棺镇妖	第二十章 鹤祖	第二十一章 血祭
第二十五章 神音如钟	第二十三章 棺中	第二十四章 对峙与选择
第二十八章 神秘果实	第二十六章 古代的星空	第二十七章 荒古禁
第三十一章 抔进虎穴	第二十九章 怒	第三十章 阿猫阿狗
第三十四章 荒之传说	第三十二章 弹指红颜老	第三十三章 韶华白首
第三十七章 灵墟洞天	第三十五章 这样的世界	第三十六章 荒古圣体
第四十一章 开辟苦海	第三十九章 生命之轮	第四十章 道经
第四十四章 仙苗	第四十二章 倒栽葱	第四十三章 神纹
第四十八章 不存于世间的古经	第四十五章 百草液	第四十六章 金色的苦海
第五十一章 一日三秋	第四十九章 良莠不齐	第五十章 玉蛇兰
第五十四章 古殿	第五十二章 深不可测	第五十三章 韩长老
第五十七章 一页金书	第五十五章 妖帝坟冢	第五十六章 心动
	第五十八章 金书镇苦海	第五十九章 妖帝之心

## Chapter page

第一章 星空中的青铜巨棺

Chapter

Setting

Bookshelf

Home

search

Login

第一章 星空中的青铜巨棺

生命是世间最伟大的奇迹。

四方上下曰宇。宇虽有实，而无定处可求。往古来今日宙。宙虽有增长，不知其始之所至。

浩瀚的宇宙，无垠的星空，许多科学家推测，地球可能是唯一的生命源地。

人类其实很孤独。在苍茫的天宇中，虽然有亿万星辰，但是却很难寻到第二颗生命源星。

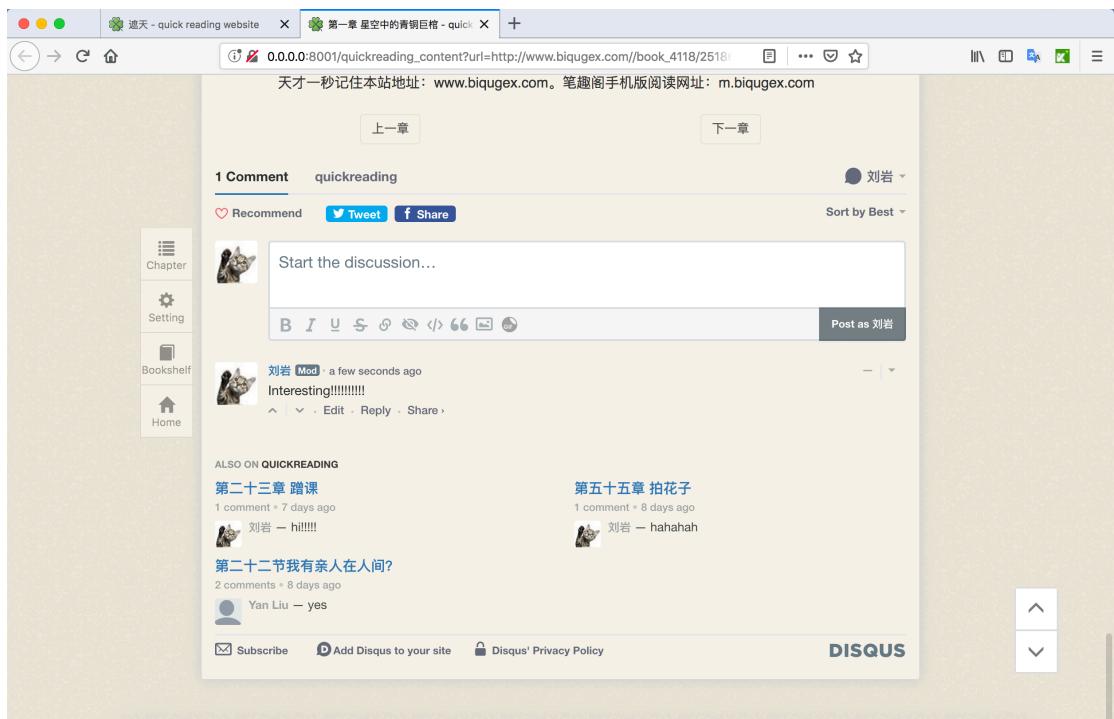
不过人类从来没有放弃过探索，自上世纪以来已经发射诸多太空探测器。

旅行者二号是一艘无人外太空探测器，于一九七七年在美国肯尼迪航天中心发射升空。

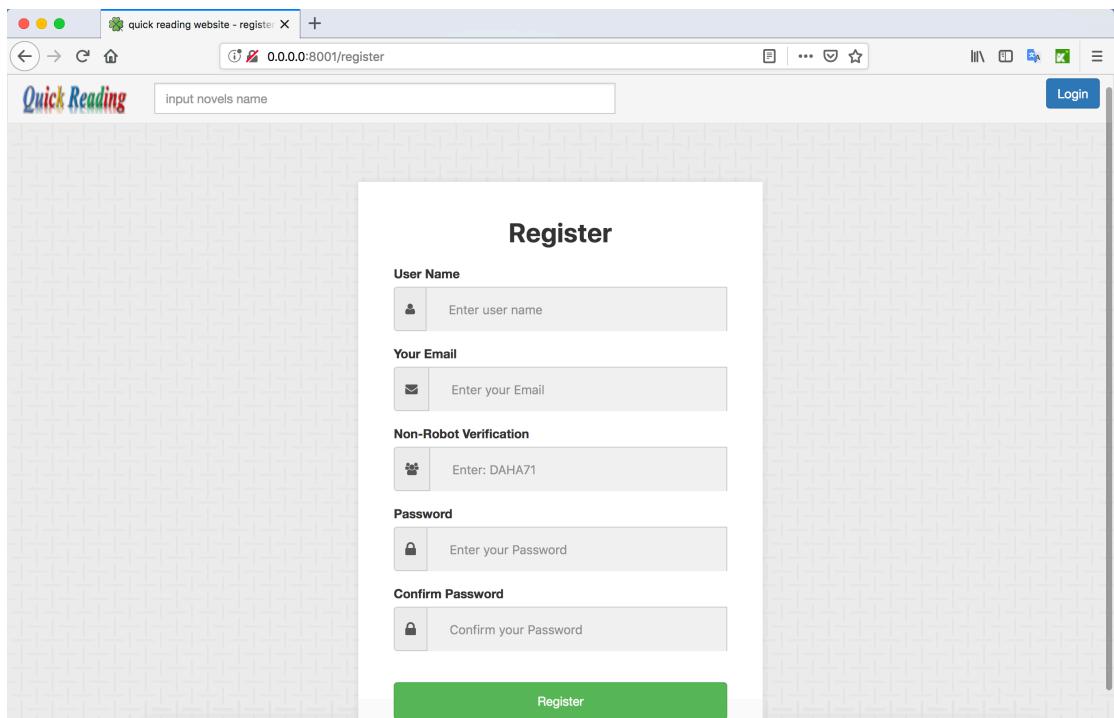
它上面携带着一张主题为“向宇宙致意”的镀金唱片，里面包含一些流行音乐和用地球五十五种语言录制的问候辞，以冀有一天被可能存在的外星文明拦截和收到。

从上世纪七十年代到现在，旅行者二号一直在孤独的旅行，在茫茫宇宙中它如一粒尘埃般渺小

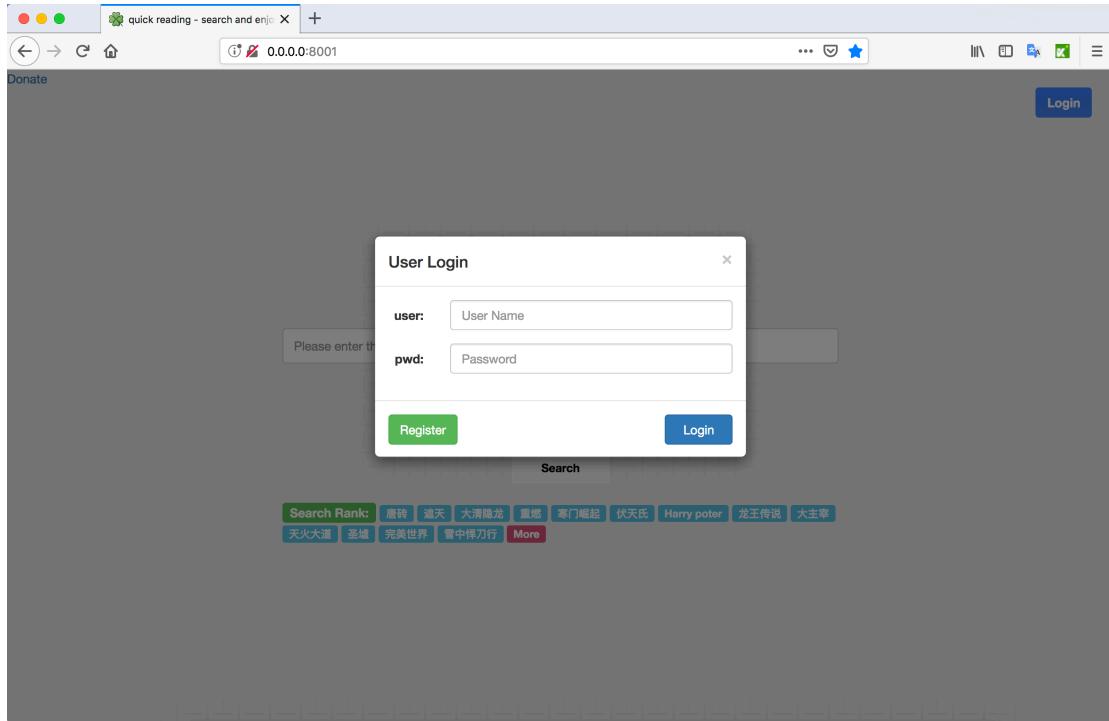
## Content page



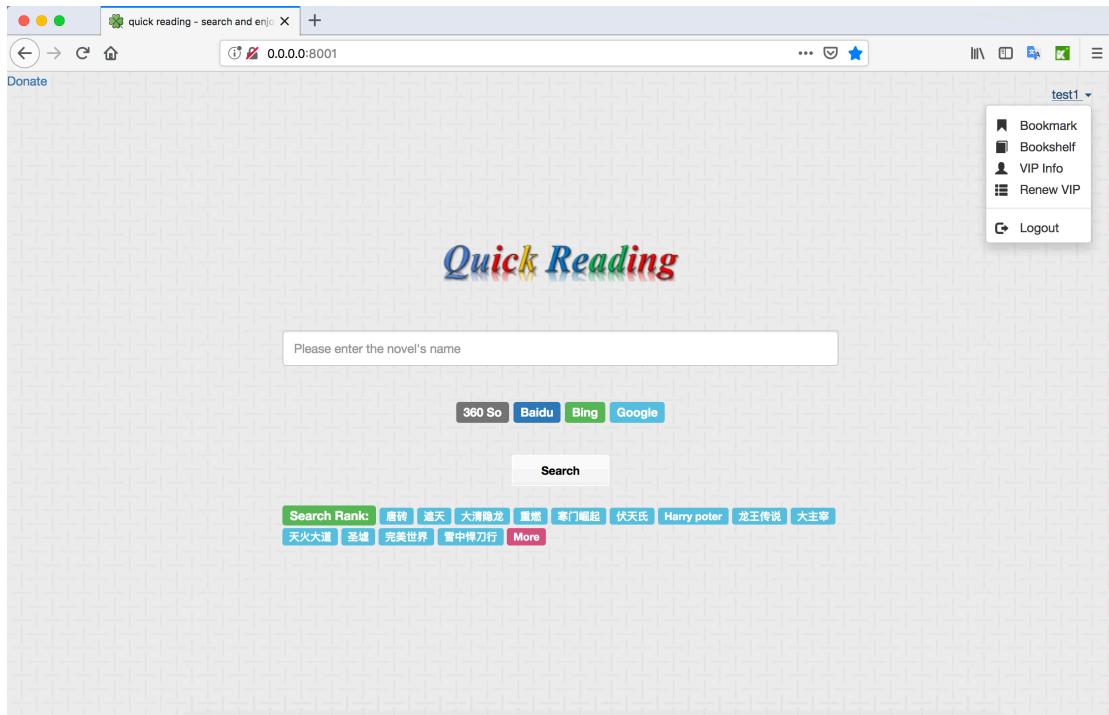
## Comment area



## Register page



Login page



User interface drop down list

Your VIP service expire date is : 2020-02-15 14:31:39

Select Type of Service

**One Month Membership**  
Only 10€

**Six Months Membership**  
Enjoy discount only 50€

**One Year Membership**  
Premium discount only 90€

**PayPal Checkout**

VISA mastercard AMEX DISCOVER

Powered by PayPal

## Payment page

Rank	Novel Name	Search Count
1	唐砖	216
2	道天	129
3	大清隐龙	103
4	重燃	88
5	寒门崛起	51
6	伏天氏	26
7	Harry poter	25
8	龙王传说	23
9	大主宰	21
10	天火大道	20
11	圣墟	18

## Searching rank page

The screenshot shows a web browser window titled "QuickReading Admin Interface" with the URL "0.0.0.0:8001/admininterface". The page displays a table of users with columns: User name, email, role, and delete. The data in the table is as follows:

User name	email	role	delete
YanLiuTest	278899085@qq.com	General User	Delete
test	278899085@qq.com	VIP User	Delete
test1	278899085@qq.com	VIP User	Delete
testtha	278899085@qq.com	General User	Delete
278899085@qq.com	278899085@qq.com	General User	Delete
testuser	278899085@qq.com	General User	Delete
test111	278899085@qq.com	General User	Delete

**The administrator manages the user pages**

## 7.0 Development Schedule

### 7.1 Gantt Chart

It shows what have been done in first semester and second semester and the process of writing development report. The first Gantt Chart shows the project completion process in the first semester. The second Gantt Chart shows how the project is developed according to the first semester's development documentation, it also show the process for writing code, testing the project, and deploying the project.

Created by Unlicensed Version



# **8.0 Problems Encountered and Resolved**

## **8.1 Python Environmental management**

Since the python version of the MAC system is 2.7, the various dependency packages I installed did not work properly, and there were many errors. So I went to Google to find a solution. Then I decided to use anaconda to update the python version of the system to 3.7 and create a python3.6 virtual environment for this project to use. Then I need to version the dependencies. I tried virtualenv at first, but it was not convenient for package management, so I give up it and choose pipenv which is the latest dependency manager for Python projects. When install pipenv, it will automatically generate two files, pipfile and pipfile.lock which control the version of packages.

## **8.2 How to use of databases in python project**

Since I only learned how to use MongoDB in Nodejs, so how to configure the database, how to connect to the database, and how to operate the database asynchronously are all challenges for me. Then I browsed MongoDB's official website and found python motor which is an asynchronous python driver for MongoDB. So, I choose it and the problem was solved according to official instructions.

## **8.3 How to get novels data from internet**

For writing crawlers, I know what frameworks and tools to use (dependency packages). However, how to establish a uniform rule to analyze different websites to obtain the novel chapters and content I need. This problem bothered me for a long time, and then I looked for a solution in Google. Eventually I found an open source crawler framework, and I contacted the author to learn how to write crawlers (howie6879, 2018).

## **8.4 Payment System**

Since I have no experience in developing payment function, I do not know what technology to use to receive payment. After Google searching investigation (Harrison, 2015) (Codecourse, 2015), I found that paypal is the easiest and fastest way. So I went to the development documentation for PayPal developer in PayPal's official website and found a way to integrate PayPal checkout on my site (Paypal, n.d.).

## **8.5 How send HTML Email**

Using Amazon's simple email service, I can easily send emails (amazon, 2019), but for emails sent from a website, it is not enough to only pursue the message delivery. The email should fully display the user name, the content of the notification message, the sender's information, and the beautiful design (including various social media contact links). But how to create HTML type variables, and make the user name in the html email change according to the user. Finally, I find the way to create html email, and because the HTML message contains CSS styles and has a lot of {}, python doesn't know where the variables are, so you need to change the {} in the CSS style to {{}} so that the variables can be passed. (stack overflow, 2018). I also find a beautiful HTML email (Witts, 2018), and change it to be used in my project.

## **8.6 Deploy problem**

How to deploy projects that require a database is a problem. Because I'm not use a cloud database, I need to find a way to include database deployment. Finally, I decided to make a mature product. Learning from stack overflow, I learn how to buy a VPS (remote server) on Vultr. The next question is how to connect to the remote server and run my project on the server. First, use the SSH command and enter the password to connect to the remote server at the terminal. Then I learned the Linux command and installed git, wget, vim and other tools. Use these tools to install MongoDB and Redis. I also need to install anaconda to configure the python environment. Clone the code from GitHub to the remote server, install the project dependencies package, and then run the project.

But, when I try to search the IP address in the browser, it still shows this webpage is not available. I was looking for this problem for a long time, finally I found the reason on CSDN blog: my project is running on port 8001, and the remote server did not open the port to people to access, after opening, it still need to add 8001 after each input IP address. I finally chose port 80 to run my project, because port 80 is the default port for IP addresses which means people search the ip address, and can see the project running on the 80 port. (chao, 2017)

For a website, it is impossible to access it through the IP address, because the IP address is hard to remember, so I choose to buy the domain name, I bought the domain name of quickreading.net in Irishdomain.com (irishdomains, 2019). However, how to connect IP address with domain name is another difficult problem. Through Google search, I know that domain name resolution is required. (alibabacloud, 2018)Finally, through aliyun domain name resolution service, IP address is successfully bound to domain name. The project was successfully deployed.

# 9.0 Self-review

## 9.1 what I learnt

The most interesting thing is that I depth learn python language. The main web programming language is JavaScript, but in my opinion, using python to write web servers is very simple and fast. Sanic server framework, in particular, supports asynchrony and makes the server respond very quickly, which is great for the user experience because people don't like to wait. In addition, python has a very powerful third-party library (dependency package), which is very developer-friendly. When I need to implement some functions, but I not know what to do, I can often find relevant methods in the library. Instead of figuring out how to do it yourself, you can save a lot of time by implementing functionality directly using existing tools. In my opinion, development projects are like producing cars, and third-party libraries are like wheels. Only put these wheels on can the car start.

Secondly, I have deepened my understanding of the database. I've only worked with relational databases like MySQL before. However, non-relational databases are becoming more and more popular, especially document storage databases like MongoDB and key value storage databases like Redis. This is also why I adopted these two databases in this project. For the use of the database, the most basic is the database configuration, when I do not know where to start, I go to the official document of the official website to learn how to install and configure these databases on my MAC (Doan, 2018). Next is how to connect to the database and store the data in the project. I find that if I want to use an asynchronous database, I need a tool (motor, asyncio\_redis, aiocahce). Thanks again to the powerful python third-party library, which gave me many solutions. In addition, it is very important for developer to design database structure. For example, in my project, the collection user\_message which stores user name, lastest\_update\_time, search\_records( contain novel name, counts of the novel), information of novels in bookshelf(contain chapter url, add\_time, reading history url), information of novels in bookmark (contain add\_time, url of chapter page).

I also have a new understanding of the web development process. Through this project, I tried the complete software development process for the first time, and the development documents (which contains functional requirements and non-functional requirements) should be determined. A formal document has a positive impact on subsequent development. A development project shouldn't just start trying to write code, developer should know what it's going to do and what it's going to do first. For example, before development, I confirmed the use of caching techniques to cache the data so that it can be directly retrieved from the cache next time. Therefore, I choose redis to cache the cookies. It helps me understand the difference between cookies and session and how to use them in website. It will initialize a session before the client request, after each

request (login, register) it will save the session. Then pass the response to client to set the client cookies

I learn a lot of techniques for python web project development. For web site development, front-end interaction is very important. I tried to completely separate the front end from the back end, but found it was not suitable for python development. It was more suitable for the development framework of Nodejs + Vue. For python website development technology, the trend is template + view + static, which means that the data is processed in the routing of the view, and processed data and some CSS styles and HTML file an JS script are sent to the template for rendering(usually use Jinjia template).

Extension of knowledge beyond code. Previously, I narrowly believed that development was just about coding. However, after the development of this project, I understood that testing and deployment are also important factors for the success of a project. I learn python unit tests by google searching and reading of official documents, and because my project involves a lot of user operation, I prefer to use automated tests. By surveying the ranking of automated testing tools, I finally chose selenium, the automated testing tool that is most popular with companies. By learning from other people's project test examples and official documents, I do automated unit testing successfully on my project. When the test passed, the deployment project was a challenge. This was the first time for me to make formal project deployment. I learned how to purchase domain name and server, configure and deploy the project by myself. With this kind of deployment, the user access is faster (compared to heroku, firebase), and my project is really a product, a website.

## 9.2 Additional features to add

Due to insufficient development time, I didn't have time to try many functions I wanted to do, such as, for a given novel, all users should be able to grade it. I also want to build a Recommendation system, users can recommend novels to other users, such as sharing a novel link.

To improve the user's mailbox authentication, I want to send notification email when user register, there is an authenticate link in the email, only users click on the link, can they register successfully (in fact, I have achieved most of the basic functions, only the link url generation and click event handling which I don't have the time to do).

In addition, the role permission control needs to be further improved, only the contents of the first few chapters are open to ordinary users, while VIP users can see all the chapters. Administrators can not only delete users, but also help users change their passwords.

# 10.0 Reference

- alibabacloud, 2018. *alibabacloud dns*. [Online]  
Available at: <https://www.alibabacloud.com/help/doc-detail/58131.htm?spm=a2796.215359.1147989.8.6cca510fljOWEo>  
[Accessed 4 2019].
- amazon, 2019. *Sending Email with Amazon SES*. [Online]  
Available at: <https://docs.aws.amazon.com/ses/latest/DeveloperGuide/sending-email.html>  
[Accessed 2 2019].
- chao, c., 2017. *Linux firewall port configuration*. [Online]  
Available at: <https://blog.csdn.net/u012674447/article/details/78890541>  
[Accessed 4 2019].
- Codecourse, 2015. *PayPal Payments: Accept Instant Payments With PayPal*. [Online]  
Available at: <https://www.youtube.com/watch?v=BD1dOWIABe0>  
[Accessed 1 2 2018].
- Doan, N., 2018. *Install and config Redis on macOS using Homebrew*. [Online]  
Available at: <https://1upnote.me/post/2018/06/install-config-redis-on-mac-homebrew/>  
[Accessed 1 2018].
- Harrison, 2015. *PayPal with Flask Web Development Tutorial*. [Online]  
Available at: <https://pythonprogramming.net/paypal-flask-tutorial/>  
[Accessed 2018].
- howie6879, 2018. *github: baidu\_novels.py*. [Online]  
Available at: [https://github.com/howie6879/owllook/blob/master/examples/novels/baidu\\_novels.py](https://github.com/howie6879/owllook/blob/master/examples/novels/baidu_novels.py)  
[Accessed 15 1 2018].
- irishdomains, 2019. *Domain and Hosting Signup*. [Online]  
Available at: [https://www.irishdomains.com/store/?NAME\\_PATH=DOMAINS\\_PATH&SCREEN=DOMAINS\\_EXTENSION\\_SCREEN](https://www.irishdomains.com/store/?NAME_PATH=DOMAINS_PATH&SCREEN=DOMAINS_EXTENSION_SCREEN)  
[Accessed 4 2019].
- Paypal, 2019. *Implement a PayPal Checkout Server Integration*. [Online]  
Available at: <https://developer.paypal.com/docs/archive/checkout/how-to/server-integration/>  
[Accessed 2 2019].
- Paypal, n.d. *Checkout Experience-paypal*. [Online]  
Available at: <https://developer.paypal.com/docs/partners/checkout/>  
[Accessed 2 2018].
- pipenv, 2018. *github: pypa/pipenv*. [Online]  
Available at: <https://github.com/pypa/pipenv>  
[Accessed 1 2018].
- Pipenv, 2018. *Pipenv: Python Dev Workflow for Humans*. [Online]  
Available at: <https://pipenv.readthedocs.io/en/latest/>  
[Accessed 1 2018].
- Real Python, n.d. *Installing Python 3 on Mac OS X*. [Online]

- Available at: <https://docs.python-guide.org/starting/install3/osx/>  
[Accessed 1 2018].
- Sanic, 2018. *Blueprint-Sanic*. [Online]  
Available at: <https://sanic.readthedocs.io/en/latest/sanic/blueprints.html>  
[Accessed 2 2019].
- stack overflow, 2018. *Sending HTML email using Python*. [Online]  
Available at: <https://stackoverflow.com/questions/882712/sending-html-email-using-python>  
[Accessed 2 2018].
- Witts, J., 2018. *Word to HTML Letter conversion*. [Online]  
Available at: <https://codepen.io/jonnowitts/pen/gZBVxL>  
[Accessed 2 2019].
- Young 、 K, 2018. *Brief analysis of the current situation of Chinese novels market*. [Online]  
Available at: <https://www.jianshu.com/p/4a93b0a7800d>  
[Accessed 2018].