A website for story collection – bookbranch web

by

Wang Ruoxi, Jack
(20300860)

Supervised by
Dr. David Rossiter

An Independent Project (CSIT 6910A; 3 Credits)

HONG KONG UNIVERSITY
OF SCIENCE AND TECHNOLOGY

March, 2015
ACKNOWLEDGEMENT

I would like to express my great gratitude towards my supervisor, Dr. David Rossiter, who had given me invaluable advice to this project.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>1</td>
</tr>
<tr>
<td>1. Introduction</td>
<td>2</td>
</tr>
<tr>
<td>2. Thesis</td>
<td>3</td>
</tr>
<tr>
<td>2.1 Use case</td>
<td>3</td>
</tr>
<tr>
<td>2.2 Database design</td>
<td>8</td>
</tr>
<tr>
<td>2.3 Laravel framework and its implementation</td>
<td>9</td>
</tr>
<tr>
<td>2.3.1 Registration and login</td>
<td>10</td>
</tr>
<tr>
<td>2.3.2 Create story description</td>
<td>11</td>
</tr>
<tr>
<td>2.3.3 Create child chapter</td>
<td>13</td>
</tr>
<tr>
<td>2.3.4 View newest description</td>
<td>14</td>
</tr>
<tr>
<td>2.3.5 View newest/previous chapter and view story line</td>
<td>17</td>
</tr>
<tr>
<td>2.3.6 Search story by title</td>
<td>20</td>
</tr>
<tr>
<td>2.3.7 View user’s information</td>
<td>21</td>
</tr>
<tr>
<td>2.3.8 Upload header</td>
<td>23</td>
</tr>
<tr>
<td>2.3.9 Follow/unfollow user</td>
<td>23</td>
</tr>
<tr>
<td>2.3.10 Alert</td>
<td>25</td>
</tr>
<tr>
<td>2.3.11 Save draft</td>
<td>25</td>
</tr>
<tr>
<td>2.3.12 Bookmarks and viewing</td>
<td>26</td>
</tr>
<tr>
<td>2.3.13 Like chapters</td>
<td>26</td>
</tr>
<tr>
<td>2.3.14 View comment notification</td>
<td>27</td>
</tr>
<tr>
<td>2.3.15 View like notification</td>
<td>28</td>
</tr>
<tr>
<td>2.3.16 View followed notification</td>
<td>29</td>
</tr>
<tr>
<td>2.3.17 View child chapter writing notification</td>
<td>29</td>
</tr>
<tr>
<td>2.3.18 Search by name</td>
<td>29</td>
</tr>
<tr>
<td>2.3.19 Block user and block story</td>
<td>31</td>
</tr>
<tr>
<td>3. Conclusion</td>
<td>32</td>
</tr>
<tr>
<td>References</td>
<td>33</td>
</tr>
<tr>
<td>Appendix</td>
<td>34</td>
</tr>
</tbody>
</table>
A website for story collection – bookbranch web

by

Wang Ruoxi, Jack

Msc in Information Technology
HONG KONG UNIVERSITY
OF SCIENCE AND TECHNOLOGY

ABSTRACT

These days’ people can interact with others on the Internet by sharing photos or videos but few of them share their idea of story writing. The bookbranch web provides a platform for people who interested in writing stories. User can write story with others, create their own favorite story based on others’ interesting ideas.
1 Introduction

The idea is about a website that has a new mechanism of story reading and creation. First title and introduction of a story can be created by one user, introduction includes characters, background and anything the user comes up with. After the story is created, any users include the creator can continue writing and create a branch of this story. When reading stories, users can choose chapters from different branches they prefer, hit the like button within chapters and even write their own branch if they got some ideas about how a different possibility can be emerge out of a specific chapter. So a story can be consisted of different branches with many chapters rank by liked times created by different users.

There are roughly two advantages of this mechanism, first, it makes writing easier and has much more fun since users don't have to write a story from the beginning to the end, they can continue and stop at any chapter they like, even create an introduction without continue writing anything, what matters here is the idea of introduction or branch is interesting or not.

Second, the mechanism brings a new experience of reading since reader can choose different version of next chapter and it makes reading story like an adventure. Also a user can choose a specific chapter and retrieve the whole story line that relates to this chapter, so reader doesn’t have to figure out a story is interesting or not by reading from the beginning, find a high quality chapter judging by its liked times and fetch the specific branch will do the trick.
2 Thesis

2.1 Use case

Diagram showing user interactions with a system, starting with a login, followed by options such as register account, create story description, create child chapters, view newest description, and a variety of other actions related to viewing and navigating stories.
Above use case is for user who haven’t login yet, use case detail are as follow

1. Login: Enter email and password for login
2. Register account: Enter email, alias, password for registration
3. Create a story description: Enter story title and description to create a new story
4. Create child chapters: Continue writing based on existing chapters or description
5. View newest description: Show all story description according to created time
6. View chapters and description according to like within a week
7. View chapters and description according to like within a month
8. View next chapter from a chapter block
9. View previous chapter from a chapter block
10. View story line according to specific chapter block: View a specific chapter and all its parent chapters, so it consists a story line
11. Search story by title
12. View other’s user information: User information includes stories this user written, stories this user liked, users this user followed and users following this user.
Along with functions that provided to user that did not login, there are some other functions that can be implemented by login user.

1. Logout
2. Upload header: Header can not exceed 200k
3. Follow/unfollow users
4. View stories liked by followed users
5. View stories written by followed users
6. Search user by name
7. Comment chapters: Can leave comments on any chapters and comments can be view by other users
8. Comment towards user: Can leave comment towards specific user who appear in comments
9. Bookmark chapters
10. Like chapters
11. View comment notifications: If other users comment this user’s stories or make comment towards this user, then he or she will receive a notification and comments can be view
12. View likes notifications: If other users liked this user’s story, then likes will be received and who liked a specific story can also be viewed
13. View followed notifications: If one user followed this user, then there will be a notification
14. View child chapter writing notifications: If this user’s story is continued by other users, then there will be a notification
15. View bookmark: Bookmark can be viewed in user main page.
16. Save draft
17. Alert

Administrator can block story from viewing by other and can block a user from login.
Database design

Database is consisted of 6 tables, relationship concepts are below

For one to many

1. user -> story
2. user -> comment
3. parent_story -> child_story
4. story -> comment

For many to many

1. user -> story (bookmark)
2. user -> user (follow relation)
2.3 Laravel framework and implementation

The website is designed by PHP framework Laravel along with JavaScript, CSS, Ajax and html 5, database is implemented by MySQL. Server side is using Ubuntu 14.04 and nignx as engine.

In Laravel framework there is an extension of html named Blade, Blade is the simple, yet powerful templating engine provided with Laravel. Unlike other popular PHP templating engines, Blade does not restrict you from using plain PHP code in your views. All Blade views are compiled into plain PHP code and cached until they are modified, meaning Blade adds essentially zero overhead to your application. Blade view files use the .blade.php file extension and are typically stored in the resources/views directory. [1]

Before user makes requests to the server, their requests will have to go through some middleware, HTTP middleware provide a convenient mechanism for filtering HTTP requests entering your application. For example, Laravel includes a middleware that verifies the user of your application is authenticated. If the user is not authenticated, the middleware will redirect the user to the login screen. However, if the user is authenticated, the middleware will allow the request to proceed further into the application. [2]

After user makes a request from a Blade, the request will be loaded to routes.php, all Laravel routes are defined in the app/Http/routes.php file, which is automatically loaded by the framework. The most basic Laravel routes simply accept a URI and a Closure, providing a very simple and expressive method of defining routes. [3]

Instead of defining all of your request handling logic in a single routes.php file, you may wish to organize this behavior using Controller classes. Controllers can group related HTTP request handling logic into a class. Controllers are stored in
the app/Http/Controllers directory. [4] inside controller there are some repositories that is responsible for database connection and SQL execution.

Below is each use case’s implementation.

2.3.1. Registration and login

Registration will need to enter a captcha code, also a validation rule is set to forbid duplicate email and user alias.

If a user fail to login 6 times, then the user’s IP will be block from login for 1 minute.
2.3.2. Create story description

Title and description are needed for create a new story, description need to be more that 50 words and title can not exceed 20 words.

If some conditions are not fulfilled, there will be notifications above the title.
Also starting a story need to fill in the captcha code.

If a user did not login, then the save method cannot be used, if it is clicked, then a notification will pop out to say that anonymous user cannot save content.

During the save process, there will be animation for saving, it is created using CSS and JavaScript.

Once the story is posting, the post button will be disable to avoid duplicate posting.

After the story is created, user will be redirect back to view newest story his story will be at the top at that time.
2.3.3. Create child chapter

This is very much the same as creating a story except there is an option beside the post button to choose whether the chapter is a normal chapter or an ending chapter.

After the chapter is posted, user will be redirected to his or her main page and see the story is posted there.
2.3.4. View newest description

The story block is shown as above, all story display here is description of stories, the upper center is the title and says it is the description, upper left number means how many times the description is viewed, one can view the whole story by clicking more or just click the content.
Above is after the content is clicked, it will show the creating time at bottom right and an alert keyword beside it.

The left bottom four key words are bookmark, comment, continue writing and next chapter, since they are all description, there will no exist precious chapter.

Only a registered user can bookmark a story or make a comment, an unregistered user can view comments but can not post one.
Every comments has its created time on the right, it display how long has it been after creating this comments, here it display 2 month ago. The time transformation is implemented using JavaScript.

When one put the mouse over a comment, it will display a reply key word, if it is clicked, then it will show reply input and button.

View according to monthly and weekly like is much the same, except it can show not only description but also chapters.

The Story display is using Scroll down effect that achieved by JavaScript. Every Scroll down will display 20 stories.
2.3.5. View next/previous chapter and view story line

Above is the story page, the upper most block is the description of the story, below is the chapter 1, if there exist multiple chapter one, there will all be shown like below test case.
And each chapters if they have next chapter, user can be choosing the story line here.

Upper left there is button, which is used to choose different chapter of a story, if a story has multiple chapter, then it will be all retrieve by clicking x in the button dropdown list.
For story list retrieving, click a story block’s title, it will retrieve its parent, the
its parent’s parent along up to the root and all display in this story page, for
example if one click the chapter 3’s title, then chapter 1-3 of this story line will
be retrieved.
2.3.6. **Search story by title**

It is implemented using SQL and it performed great by only search relevant title.
2.3.7. View user’s information

The upper four buttons means wrote, liked, following and followed.
Above are all functions for non-registered user, below is functions for registered user.
2.3.8. Upload header

After click the header of the registered user’s own main page, there will be a pop out window for image uploading.

If the image size is larger than 200k or it is not a image form, then the upload will fail along with a alert pop out by JavaScript.

2.3.9. Follow/unfollow user

After the user is login, the follow/unfollow button will shown as below
In the process of database manipulation, once a user wants to follow another user, first the JavaScript will test whether the user is clicking a follow button or an unfollow button, then it will send the correct request to the server, whether display follow or unfollow is judged in Blade. But it can still cause mistakes, so if the server has the request that A is following B, then it will first check whether in the follow table that A is truly not following B, if it is true, then increment B’s followed number and A’s following number, then send back the respond that the operation is succeed, JavaScript received the respond and turn the follow button into unfollow button.

If a user clicks a follow button multiple times, it will only trigger one request until the first request has responded. It is handled in Ajax and JavaScript.

Also the transaction is set, if any process is wrong in the server, then it will roll
back all SQL and direct user to a page says there is a error, all post requests are using transaction handing.

2.3.10. Alert

Only a register user can click alert, then it will make administrator know that a story is alerted.

2.3.11. Save draft

Once a draft is saved, the next time user click new story or continue writing, it will shows the same draft. And if user posted the saved draft, it will be erased. since the website is using bootstrap, it can view in mobile phone, saving method makes it convenient for writing story on mobile side.
2.3.12. Bookmarks and viewing

After a story is bookmarked by current user, it will show cancel and it will become blue. Again click it will cancel the bookmark.

Bookmark is viewed in one’s main page by click the button below his or her own header, in other user’s main page the button is follow button.

2.3.13. Like chapters

If a chapter is liked by current user, then the number will become blue, otherwise there is no color.

The process of liking a chapter is very much the same as follow/unfollow a user, the effects are achieved using Ajax and CSS, JavaScript.
2.3.14. View comment notification

Once a notification is occurred, the it will show in the upper bar as a number and detail will show what kind of notification it is.

![Notification Icon]

- **View comment notification**
  - Once a notification is occurred, it will show in the upper bar as a number and detail will show what kind of notification it is.

---

**About Book**

- **Overview**

**User Comments**

- **User 1**
  - March 24
  - Comment: 被关注

- **User 2**
  - April 1
  - Comment: 我也想知道这个网站来着

----

**About User**

- **User 1**
  - March 24
  - User ID: zcuban

- **User 2**
  - April 1
  - User ID: 北极光

---

**User Actions**

- **User 1**
  - Followed

- **User 2**
  - Message

---

**User Information**

- **User 1**
  - User ID: Jack Wang
  - Date: March 24

---

**User Profile**

- **Profile 1**
  - User ID: zcuban

- **Profile 2**
  - User ID: 北极光

---

**User Activities**

- **Activity 1**
  - User ID: Jack Wang
  - Date: March 24
  - Action: Followed

---

**User Feedback**

- **Feedback 1**
  - User ID: zcuban
  - Date: March 24
  - Comment: 被关注

- **Feedback 2**
  - User ID: 北极光
  - Date: April 1
  - Comment: 我也想知道这个网站来着
Once user’s story is commented or some one is commenting this user in other story, it can be view in comment notification. Click the title will get the story line of the story. Click user name or header to go to the user’s page. If the comment is new, the background will be blue.

2.3.15. View like notification

<table>
<thead>
<tr>
<th>姓相</th>
<th>第1章 梦</th>
<th>17 天前</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>神的国度</th>
<th>简介</th>
<th>25 天前</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>虚拟与真实</th>
<th>简介</th>
<th>1 个月前</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Likes notification will show stories that are liked by how many users, and its users’ detail.

If the likes is new, then the button will be blue.
If the likes is new, the user’s background will be blue.

2.3.16. View followed notification

If this user is followed by other users, then he or she will be notified. Click the notification will direct the user to main page to see the user who is newly following this user.

2.3.17. View child chapter writing notification

If a user’s story is continued by other, then the user will be notified. Newly writing story’s background will be blue.

2.3.18. Search user by name

After login, user can search both story and user, user can be search by name.
Above are all functions for registered user, below is functions for administrator.
2.3.19. Block user and block story

Administrator has privilege to block story and user by click the admin button.
In this page administrator can view story which is alerted and its author, after viewing the story, administrator can choose to block the story by click the block button on the upper right, below is the number of times of alerting, on the right is it author and how many times the user’s all story is alerted, the button below is used to block the user.

**Conclusion**

This paper is focusing the idea’s implementation with Laravel, it implemented quite well and after testing the website is stable, it used lazy loading to keep the SQL generation to be a constant number, now the website has gain its users on the internet, so in the future it may be able to analysis data from this website.
References

[1] https://laravel.com/docs/5.2/blade


Appendix

Minutes of the 1st Project Meeting

Date: Tuesday, 1 March 2016

Time: 4:20 PM

Place: Room 3512

Attending: Prof. Rossiter, Jack Wang

Absent: None

Recorder: Jack Wang

Approval of minutes

This is first formal group meeting, so there were no minutes to approve.

Report on Progress

Discussed the idea and settle the plan

Discussion Items and Things To Do

How to implement and framework to use

Meeting adjournment

The meeting was adjourned at 4:40 AM.

Minutes of the 2nd Project Meeting

Date: Friday, 18 March 2016
Time: 4:20 PM

Place: Room 3512

Attending: Prof. Rossiter, Jack Wang

Absent: None

Recorder: Jack Wang

Approval of minutes

The minutes of the last meeting were approved without amendment.

Report on Progress

Website is built in its first version

Discussion Items and Things To Do

Extends some function such as searching

Meeting adjournment

The meeting was adjourned at 11:40 AM.

Minutes of the 3rd Project Meeting

Date: Thursday, 24 March 2016

Time: 10:30 PM

Place: Room 3512

Attending: Prof. Rossiter, Jack Wang
Absent: None

Recorder: Jack Wang

Approval of minutes

The minutes of the last meeting were approved without amendment.

Report on Progress

Website is built in its second version

Discussion Items and Things To Do

Extends some function such as tree structure

Meeting adjournment

The meeting was adjourned at 10:50 AM.

Minutes of the 4th Project Meeting

Date: Friday, 15 April 2016

Time: 2:30 PM

Place: Room 3512

Attending: Prof. Rossiter, Jack Wang

Absent: None

Recorder: Jack Wang
Approval of minutes

The minutes of the last meeting were approved without amendment.

Report on Progress

Website is built in its Third version

Discussion Items and Things To Do

Extends some function such as specifying the tree structure

Meeting adjournment

The meeting was adjourned at 2:50 AM.