Douban Sauce Website

AUTHORS

Xingxing Liu:

Yue Lyv:

Wen Qin: qin.w@husky.neu.edu

# ABSTRACT

This report is to give a brief description of a website named Douban Sauce Website (DSW). DSW provides a platform for users to exchange house information and transact real estate.

Chapter 2 introduces DSW briefly. Chapter 3 to 5 explains DSW in details, including the requirements, website design and implementation. Chapter 6 displays the instructions to use the website. Chapter 7 concludes the project work.

# INTRODUCTION

There are three types of user on the website, including administrator, corporation user and individual user. Both of corporation user and individual user have right to create, edit and delete his own house/apt post, in which basic information is needed, including address, housing area, price, purpose, etc. They could also give rating score and make comments on house/apt post, and post rating score and reviews for other users. All of three types are able to use the webpage to search for information by filters. More descriptions are as followed.

- Administrator

Have right to post announcement. Have access to review, edit and delete posts by other types of user.

- Corporation User

House/apt brokers, tagger as “provider”. Provide basic corporation information and contact information.

- Individual User

Could be House/Apt owners or seekers. The role of the individual user is set default as “seeker”. When a user makes a post, a tag “provider” will be added to this user.

The information supplied on the website includes:

Information about individual user or corporation user.

Rating score and reviews for individual user or corporation user.

Basic information of the house posted.

Rating score and reviews for each house posted.

# REQUIREMENTS

## System

### Douban Sauce Website [DSW]

|  |
| --- |
| Name: Douban Sauce Website |
| Description: The DSW as a participant in use cases. All communication between any user and the DSW is assumed to occur on an authenticated and secure channel. Activities such as login are therefore not included in any of the use cases. The term item is used generically for any entity that is administered by the DSW with versioning and annotations such as provenance information. This includes modules, mappings, configurations, compositions, process definitions and policies. Configuration is itself a generic term that includes contexts, frameworks and situations. |

## System Actors

### Administrator [ Administrator ]

|  |
| --- |
| Description: A top-level registration authority. An administrator is a kind of user with specific capabilities and responsibilities that are not common to all users. |
| KindOf: #IndividualUser |

### Corporate User [ CorporateUser ]

|  |
| --- |
| Description: An house broker corporate that creates housing posts, reviews and ratings for other posts or [#IndividualUser]. The [#Administrator] verifies who can be a corporate user. |
| KindOf: #IndividualUser |

### Individual User [ IndividualUser ]

|  |
| --- |
| Description: An individual, either a house owner or a seeker. The individual user creates housing posts, reviews and ratings for other posts or [#IndividualUser]. The individual user is changed to [#CorporateUser] after submitting corporate information and verified by [#Administrator]. |
| KindOf: #User |

### General User [ User ]

|  |
| --- |
| Description: A user of the DSW. This includes registered users as well as individuals who are searching the DSW or retrieving from it in an ad hoc manner. Unlike registered users, users in general need not go through the process of registration. |

## Use Cases

### Individual User Register [ individualRegister ]

|  |
| --- |
| Description:  A user wants to register to become an Individual User. |
| Step-by-step Description:   1. [#User] A user requests that a new registration be admitted. 2. [#DSW] The system checks whether the username exists or not. If not, the registration will succeed, otherwise fail. The system notifies the user the result. |

### Corporate User Register [ corporateRegister ]

|  |
| --- |
| Description:  An individual user wants to register to become a Corporate User. |
| Step-by-step Description:   1. [#IndividualUser] The user uploads required materials and submits the request to be a corporate user. 2. [#DSW] The system notifies the [#Administrator] that a request has been made for an individual user to become a corporate user. 3. [#Administrator] The administrator verifies the material and approves or rejects the request. 4. [#DSW] The system notifies the user the result. |

### Query the Registration authorities [ queryAuthorities ]

|  |
| --- |
| Description:  A user queries the system to obtain information on relevant registration authorities. This will only retrieve publicly released information about the authorities. |
| Step-by-step Description:   1. [#User] A user requests information on registration authorities. 2. [#DSW] The system returns the information about all relevant registration authorities of the specified user. |

### Change View Mode [ changeMode ]

|  |
| --- |
| Description:  A user requests to change the mode of the result view. This request will inform the front website to use a different view frame with the default information. The view mode includes list-view, big-image-view, or small-icon-view. |
| Step-by-step Description:   1. [#User] The user submits the change view mode request. 2. [#DSW] The front website changes the view frame and show the default information on the website. |

### Reset Password [ resetPassword ]

|  |
| --- |
| Description:  A user requests to change the password of his/her own user account by query for the password information out from the system. This will only change the current user’s password and only password. |
| Step-by-step Description:   1. [#IndividualUser] The user submits the change password request. 2. [#DSW] The system returns information about the password of current user. 3. [#IndividualUser] The user types in a new password and submits it to the system. 4. [#DSW] The system verifies user and stores new password. The system notices the user Reset Password successfully. |

### Update Personal Information [ updateInformation ]

|  |
| --- |
| Description:  A user queries for personal information and update it. Personal information includes account name, email address, phone number, account id. |
| Step-by-step Description:   1. [#IndividualUser] The user submits query about personal information. 2. [#DSW] The system returns a personal information form including all the personal information. 3. [#IndividualUser] The user makes changes on it and sends it back to system. 4. [#DSW] The system replaces original information with updated one. |

### Query My Reviews [ queryReview ]

|  |
| --- |
| Description:  A user queries for the reviews posted by current user. The system will return information about the reviews. |
| Step-by-step Description:   1. [#IndividualUser] The user submits the request for querying the user’s reviews. 2. [#DSW] The system returns the information about the reviews posted by current user’s account id. |

### Query My Posts [ queryPost ]

|  |
| --- |
| Description:  A user queries for the posts posted by current user. The system will return information about the posts. |
| Step-by-step Description:   1. [#IndividualUser] The user submits the request for querying posts. 2. [#DSW] The system returns the information about the posts posted by current user’s account id. |

### Filter Search [ filteredSearch ]

|  |
| --- |
| Description:  A user queries for posts by different attributes, such as combination of house type, year, location, No. of bedroom, price. The system will return information about the posts. |
| Step-by-step Description:   1. [#User] The user submits the query for rental information by submitting a combination of attributes. 2. [#DSW] The system returns a list of posts that match the query. |

### Sorted Search [ sortedSearch ]

|  |
| --- |
| Description:  A user submits a request for sorting the search result ordered by attributes, such as by year, by prices, by house type, and by number of bedrooms. |
| Step-by-step Description:   1. [#User] The user submits the request for sorting the search results by specified attributes. 2. [#DSW] The system returns the sorted list of posts. |

### New Post [ newPost ]

|  |
| --- |
| Description:  A User creates a new post for a new house. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a form with information of house. 2. [#DSW] The system puts the request queued, waiting for verification, and sends a notification to the user. |

### Edit Post [ editPost ]

|  |
| --- |
| Description:  A User updates a posts. |
| Step-by-step Description:   1. Include: [#queryPost] 2. [#IndividualUser] The user submits a form with updated information of house. 3. [#DSW] The system puts the request queued, waiting for verification, and sends a notification to the user. |

### Delete Post [ deletePost ]

|  |
| --- |
| Description:  A User deletes a posts. |
| Step-by-step Description:   1. Include: [#queryPost] 2. [#IndividualUser] The user submits a delete request. 3. [#DSW] The system re-confirms with the user. 4. [#IndividualUser] The user submits the request by confirming, or cancels the request by cancelling. 5. [#DSW] The system removes the record from the database if the user confirms the delete request. The system returns a notification to the user. |

### Query Post [ queryPost ]

|  |
| --- |
| Description:  A User queries the information of a post. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a query for a post. 2. [#DSW] The system returns the post. |

### Verify Post [ verifyPost ]

|  |
| --- |
| Description:  The administrator verifies the new post request or updated post request. |
| Step-by-step Description:   1. [#Administrator] The user gets a list of unverified posts. 2. include: [#queryPost] 3. [#Administrator] The user verifies a post. 4. [#DSW] The system changes VerificationStatus of the post, and sends a notification to the user. |

### Get Seller Information [ getSellerInfo ]

|  |
| --- |
| Description:  A User wants to get the a seller’s information. |
| Step-by-step Description:   1. [#User] The user submits a request of getting information of a seller by username. 2. [#DSW] The system returns the requested information. |

### Top Post [ topPost ]

|  |
| --- |
| Description:  An Administrator makes a post to the top of the webpage. |
| Step-by-step Description:   1. include: [#queryPost] 2. [#Administrator] The user submits a request of making a post to the top of the webpage. 3. [#DSW] The system changes topping information, and sends notification to the user. |

### User Rating [ userRating ]

|  |
| --- |
| Description:  A User posts a rating for another User. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a rating for another user by username. 2. [#DSW] The system records the rating and update the rating average of the specified user. |

### Post Rating [ houseRating ]

|  |
| --- |
| Description:  A User posts a rating for a housing post. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a rating for a housing post. 2. [#DSW] The system records the rating and update the rating average of the specified post. |

### Post Views [ postViews ]

|  |
| --- |
| Description:  A User requests view times of a post. |
| Step-by-step Description:   1. [#User] The user requests the view times of a housing post by postid. 2. [#DSW] The system returns the view times of the specific post. |

### Post Average Rating [ postAveRating ]

|  |
| --- |
| Description:  A User requests an average rating of a post. |
| Step-by-step Description:   1. [#User] The user requests the average rating of a housing post by postid. 2. [#DSW] The system returns the average rating of the specific post. |

### User Average Rating [ userAveRating ]

|  |
| --- |
| Description:  A User requests an average rating of a user. |
| Step-by-step Description:   1. [#User] The user requests the average rating of a user by username. 2. [#DSW] The system returns the average rating of the specific user. |

### Post Tags [ postTags ]

|  |
| --- |
| Description:  A User requests all tags added to a post. |
| Step-by-step Description:   1. [#User] The user requests all tags added to a post by postid. 2. [#DSW] The system returns all tags added to the specific post. |

### Add Tag [ addTag ]

|  |
| --- |
| Description:  A user adds a tag to a housing post. A tag describes the house, such as “fine decoration”, “spacious”. |
| Step-by-step Description:   1. Includes: [#postTags]. 2. [#IndividualUser] The user submits a new tag. 3. [#DSW] The system updates the tag statistics. |

### Create Review [ createReview ]

|  |
| --- |
| Description:  A user wants to add a review to a housing post. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a new review for a housing post. 2. [#DSW] The system returns a confirmation message. |

### Query Review [ queryReview ]

|  |
| --- |
| Description:  A user wants to see a review to a housing post. |
| Step-by-step Description:   1. [#User] The user submits a request of the information of a review. 2. [#DSW] The system returns the requested review. |

### Edit Review [ editReview ]

|  |
| --- |
| Description:  A user wants to edit a review to a housing post. |
| Step-by-step Description:   1. Include: [#queryReview] 2. [#IndividualUser] The user submits an update of a review for a housing post. 3. [#DSW] The system updates the review. |

### Delete Review [ deleteReview ]

|  |
| --- |
| Description:  A user wants to delete a review to a housing post. |
| Step-by-step Description:   1. Include: [#queryReview] 2. [#IndividualUser] The user submits a delete request of a review for a housing post. 3. [[#DSW]](http://www.ccs.neu.edu/home/kenb/ontologies/oor-usecase.xml#OOR) The system deletes the review. |

### Send Email [ sendEmail ]

|  |
| --- |
| Description:  A user wants to send an email to other users. |
| Step-by-step Description:   1. [#IndividualUser] The user inputs one or more usernames to “Receipt”, edits subject and content, and then submits request to send out the email to all the specified users. 2. [#DSW] The system sends the requested email and returns a confirmation message. |

### Receive Email [ receiveEmail ]

|  |
| --- |
| Description:  A user wants to see all the emails in DSW inbox. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a request to receive all the emails in DSW inbox. 2. [#DSW] The system returns the list of requested emails. |

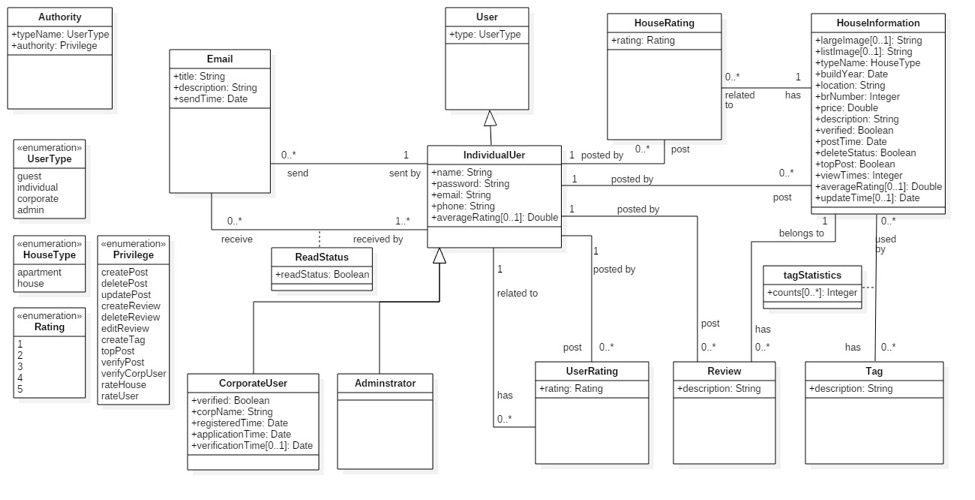
### Query Email [ queryEmail ]

|  |
| --- |
| Description:  A user wants to query an email. |
| Step-by-step Description:   1. [#IndividualUser] The user submits a request for an email. 2. [#DSW] The system returns the specific email. |

### Send Notification [ sendNotification ]

|  |
| --- |
| Description:  The administrator wants to send a notification to all the individual users. |
| Step-by-step Description:   1. [#Administrator] The user edits subject and content, and then submits a request to send out the notification to all the individual users. 2. [#DSW] The system sends out the requested email and returns a confirmation message. |

# DESIGN



explain user relationships

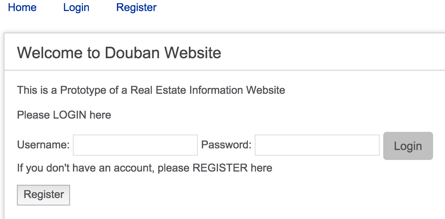
# IMPLEMENTATION-Yue

Describe the implementation

# DISCUSSION

## User Login

An existing user logs in the system on the following page. After filling out “username” and “password”, a user logs in by clicking “login” button.



# CONCLUSION

Our team established the DSW which supplied a platform for users to exchange house information and make transaction of real estates.

The implementation of database was in accordance with UML design. The website covered all the system actors and user cases designed in the requirements.

But there is still some work to do, including the embellishment of User Interface, replacement of synchronous request with asynchronous request from database.

# REFERENCES

[1] Edward Sciore. (Oct. 2008). *Database Design and Implementation*. John Wiley.

[2] Ken Baclawski*. NU CCIS Ontology Web Site*. Retrieved Oct 19, 2015, from http://www.ccs.neu.edu/home/kenb/ontologies/index.html

[3] British Columbia Institute of Technology. (Oct 31, 2015). *CodeIgniter User Guide*. Retrieved Nov 2, 2015, from http://www.codeigniter.com/user\_guide/