# Reflective report

Yang Tianyi

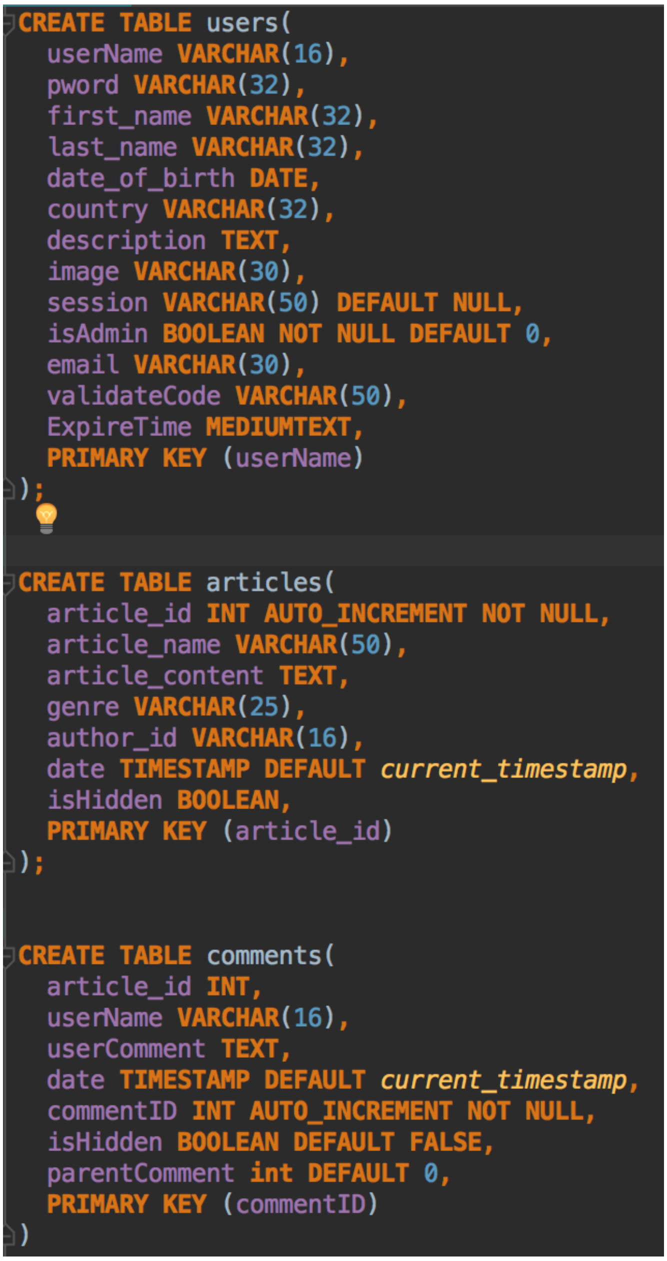
tyan227 644294325

The goal of the project is creating a blogging website. The basic functionality of a blogging system consists of many sections, for example: users can register for an account, which is needed to be able to post articles and to leave comments on others. When logged in, they have full control of the content they have authored: creating, updating and deleting their content and comments. For the advanced features, we choose Pack One, Two and Three for this project, which means that we have implemented features like administrative/user interface, hiding and showing of articles, reCAPTCHA, search and sorting.

The following aspects are mainly taken into consideration in the design of the system.

1. Database:

All of the information mentioned above should be saved into a database, which is mainly divided into 3 sections, containing tables of users, articles and comments. The details of each table is shown below:



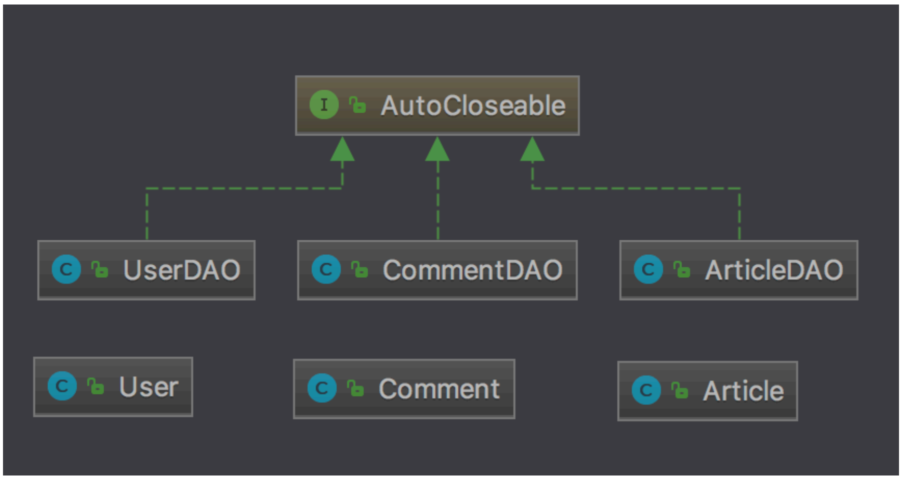
```
CREATE TABLE users(
  userName VARCHAR(16),
  pword VARCHAR(32),
  first_name VARCHAR(32),
  last_name VARCHAR(32),
  date_of_birth DATE,
  country VARCHAR(32),
  description TEXT,
  image VARCHAR(30),
  session VARCHAR(50) DEFAULT NULL,
  isAdmin BOOLEAN NOT NULL DEFAULT 0,
  email VARCHAR(30),
  validateCode VARCHAR(50),
  ExpireTime MEDIUMTEXT,
  PRIMARY KEY (userName)
);

CREATE TABLE articles(
  article_id INT AUTO_INCREMENT NOT NULL,
  article_name VARCHAR(50),
  article_content TEXT,
  genre VARCHAR(25),
  author_id VARCHAR(16),
  date TIMESTAMP DEFAULT current_timestamp,
  isHidden BOOLEAN,
  PRIMARY KEY (article_id)
);


CREATE TABLE comments(
  article_id INT,
  userName VARCHAR(16),
  userComment TEXT,
  date TIMESTAMP DEFAULT current_timestamp,
  commentID INT AUTO_INCREMENT NOT NULL,
  isHidden BOOLEAN DEFAULT FALSE,
  parentComment int DEFAULT 0,
  PRIMARY KEY (commentID)
)
```

2. Hikari Connection Pool:

Hikari is the ideal solution to make connection to the database, which allows users to retrieve connections when required - creating fresh connections if necessary or reusing existing open connections. It can also be configured to keep connections open if required, and to maintain more than one connection for situations where user may need several at once.

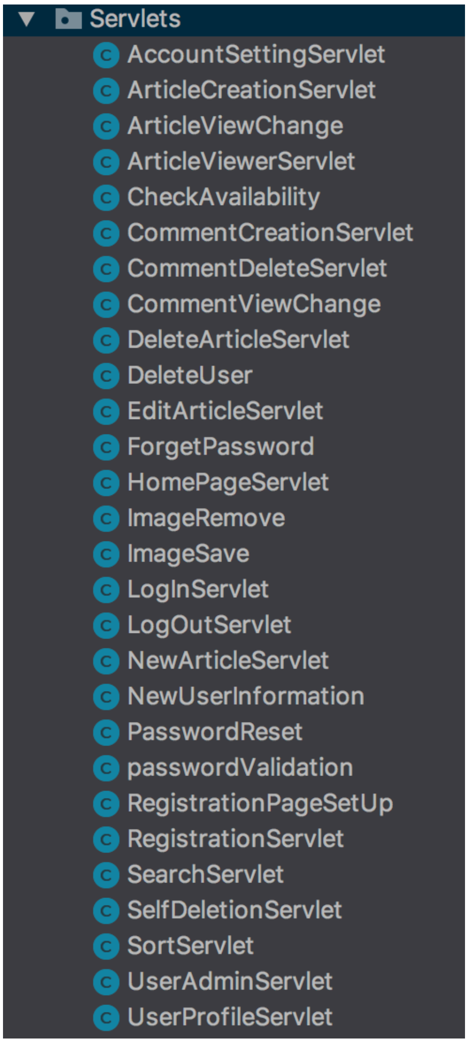
3. POJO and DAO:

For the final project, the data stored in the database is huge. So for an easier and more convenient access to the database, POJO and DAO are included in the project, which makes the information of the database more intuitive, and in this way the approach to the data using java is more in line with object thinking. After the construction of these elements mentioned above, the foundation of the project is completed.



4. Servlet:

Servlet is the core for the project to realize all of the dynamic features of the blogging website. Basically, the functions of the blog can be identified from each individual servlet listed below:



```
▼ Servlets
    AccountSettingServlet
    ArticleCreationServlet
    ArticleViewChange
    ArticleViewerServlet
    CheckAvailability
    CommentCreationServlet
    CommentDeleteServlet
    CommentViewChange
    DeleteArticleServlet
    DeleteUser
    EditArticleServlet
    ForgetPassword
    HomePageServlet
    ImageRemove
    ImageSave
    LogInServlet
    LogOutServlet
    NewArticleServlet
    NewUserInformation
    PasswordReset
    passwordValidation
    RegistrationPageSetUp
    RegistrationServlet
    SearchServlet
    SelfDeletionServlet
    SortServlet
    UserAdminServlet
    UserProfileServlet
```

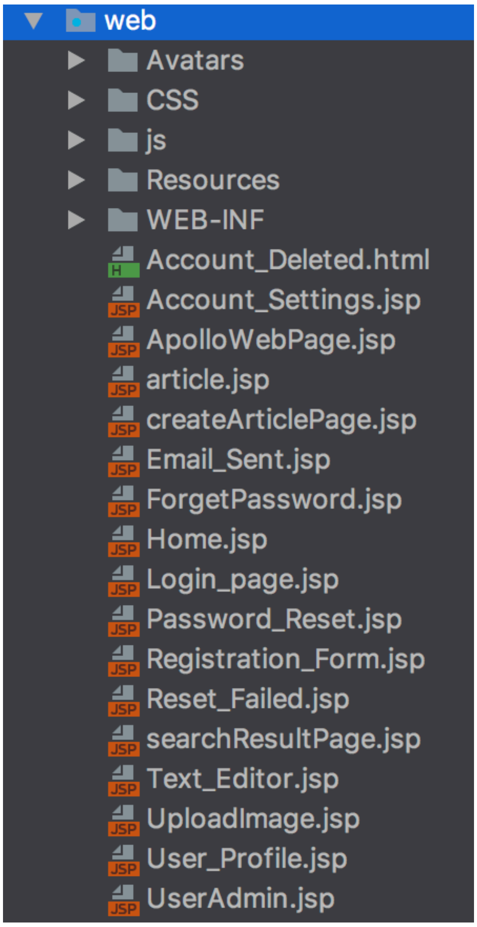
The relationship between servlets and web pages is very close. On the one hand, the data submitted by the user (usually by form) through the web page is received on the servlet and further processed. On the other hand, after proper processing, the data will be displayed on a result web page. The user may seem to be interacting with different web pages, but as a matter of fact, through servlet, user is actually reading or saving data from the database.

In order to retrieve correct information, the setting of attributes and parameters (the name attributes of each input from a form) should be corresponding to each other.

5. Web pages:

Web pages are crucial for blog sites because the product finally presented to the user is not the code but the web pages. The appearance of the web pages directly defines the users' first impression, and the performance of the blog will effectively affect the users' experience and evaluation.

Each ideas of the features of the final project comes from the actual website. Then all these ideas break into executable codes. In the end, codes will be merged and reassembled back to the website to fulfill all the requirements.



```
▼ web
    ▶ Avatars
    ▶ CSS
    ▶ js
    ▶ Resources
    ▶ WEB-INF
      Account_Deleted.html
      Account_Settings.jsp
      ApolloWebPage.jsp
      article.jsp
      createArticlePage.jsp
      Email_Sent.jsp
      ForgetPassword.jsp
      Home.jsp
      Login_page.jsp
      Password_Reset.jsp
      Registration_Form.jsp
      Reset_Failed.jsp
      searchResultPage.jsp
      Text_Editor.jsp
      UploadImage.jsp
      User_Profile.jsp
      UserAdmin.jsp
```

You may notice that most of the pages in our project are JSP files. This is because web pages usually need to obtain servlet-processed data and present to the user with different results under different conditions.

6. JavaScript:

Through servlets and JSP, we can easily modify the web page using java code. However, in some cases, we still need JavaScript to achieve a more intuitive and more convenient implementation of specific features. For example, in this project, user avatar snipping and username real-time repetition detection are all achieved through JavaScript.

For this project, I mainly contributed the following aspects to my team:

1. Selecting icon for the Apollo blogging website.

2. Users are able to upload an image from their computer, from which a thumbnail can be generated by zooming and snipping for use as a custom avatar.

3. When logged in, users are able to comment on articles. Users are also able to delete any comments they have written, as well as any comments on articles which they have authored.

4. Users are able to edit any of their account information, and also be able to delete their account.

5. Administrator can reset user passwords, by sending the corresponding user an email with a link to reset their password. User can also reset their password if they forget it.

6. Informing new users when creating a new account or old users when modifying profiles whether or not their chosen username is already taken in real time, without having to submit a form.

7. Users are able to sort article lists by article title, username, genre and date.

8. Reconstruction of the database to realize advanced features.

9. Reconstruction of the POJOs and DAOs.

10. Build connection between Servlet and corresponding web pages.

When I received the handout of the final project, I felt stressed because it seemed to be very challenging and complicated. However, after patient analysis and group discussion, it was found that most of these features were taught and explained in class. Details of these topics taught in class is briefly explained below:

Backend:

All the information of users, articles and comments, is stored in the database.

In order to get connect to the database, we use Hikari connection pool.

To pass the data in as an object, POJOs and DAOs are used for this project.

Servlets are used to update information displayed on the websites.

Frontend:

Html is used for web pages, and in order to set conditions for displaying specific content, a large number of JSP files are widely used for this project.

CSS and Bootstrap are widely used for the website to have an appearance good enough to attract our users.

In order to realize functions like avatar snipping, JavaScript also plays an important role in this project.

At the same time, when we tried to implement some of the features from the handout, we found out that there were also some of the contents that was first appeared in the final project. Although no corresponding guidance could be found in the lectures, we manage to realize them by looking through relative content from the Internet.

In compulsory features basic functionality number two, users must also be able to upload an image from their computer, from which a thumbnail can be generated for use as a custom avatar. We have leant how to upload photo using java during the lab. But when we select a photo that is not of a perfect square shape, the image may seem stretched on horizontal or vertical sides. Many websites have a feature allowing users to cut image and upload it as their avatar, which is a perfect solution for this problem, so I decided to implement a similar feature to the website.

The crop feature works perfect using JavaScript. It’s a well received function so that many guide lines could be found explaining in detail the functionality of it. By adding cropbox.js and jQuery as library files, making some corresponding modifications for a consistent view and building connections to the websites and servlet, this function managed to get installed on the website.

In advanced features pack one: administrative interface, the administrator should be able to reset user passwords, by sending the corresponding user an email with a link to reset their password. Actually there are two difficulties in this one sentence:

1. Send email by Java:

To send an e-mail using Java Application requires Java Mail API and Java Activation Framework (JAF) get added as libraries for the project. Then, various aspects of the email should be set correctly:

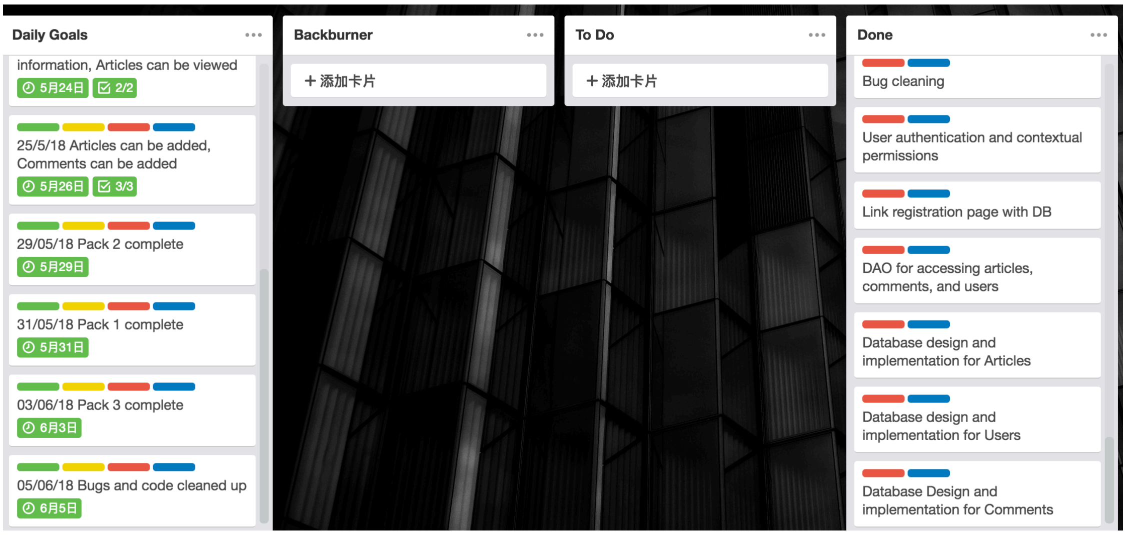
- The host of the email with the port, which may affect the result of sending emails when the project is deployed.
- The email address and password (or an authentication code, which is of higher security level).
- The head and body of the email.

In this way, the main function of the email is completed, and for this project, the only thing left is to pass in the username, user's email address and password reset link.

2. Generate the link, and send data to the servlet:

After referring to relevant topics from Internet, I realized that the attributes are actually inside the link. For example, in the following link:
http://localhost:8080/passwordValidation?sid=333b065facdabd47e75479248ec665bf&userName=qqq
we can easily get the parameters of "sid" and "userName", which can be used for the servlet. From the example, we know that the base path and username for a link is very likely to repeat. So in order to avoid generating a same link, the signature (sid) should be always different. For this project, the signature consists of 3 elements: username, date and a secret key which is a random number. Because the signature contains these personal information, it should be encrypted for security concern. In this project, I use MD5 crypt method, which will keep the link unique and hard to decrypt. Also, this link is only valid for 24 hours, which prevents the user to reset the password over and over later on.

Working as a team for this project is really a memorable experience for me. The benefit for a team is, in order to achieve the goal of the project, everyone fought to contribute to the team, trying to share more responsibility in order to make the project proceed smoothly. In this state of mind, we accomplished the tasks as planned for each day with great effort. In fact, we don't spend too much time in the lab every day, but the work for each of us is fulfilling. We also use the rest of the day to conceive the whole project and planning the next day's work. During the project, the best moment for a day is to move a task from “To Do” list to “Done” in Trello.



Daily Goals

information, Articles can be viewed
5月24日 ☑ 2/2

25/5/18 Articles can be added, Comments can be added
5月26日 ☑ 3/3

29/05/18 Pack 2 complete
5月29日

31/05/18 Pack 1 complete
5月31日

03/06/18 Pack 3 complete
6月3日

05/06/18 Bugs and code cleaned up
6月5日

Backburner

＋添加卡片

To Do

＋添加卡片

Done

Bug cleaning

User authentication and contextual permissions

Link registration page with DB

DAO for accessing articles, comments, and users

Database design and implementation for Articles

Database design and implementation for Users

Database Design and implementation for Comments

The difficulty is mainly caused by the merging of the project. In order to avoid that, we tried to divide the project into independent tasks for everyone. And everyday before we started the new tasks, we would merge all the work done by the previous day together, and pushed it to Bitbucket. And then everyone pulled the latest version of it, and used it as the start point of the day. In this way, a great amount of conflicts is avoided, which guarantees our work with high efficiency.

I would like to thank all my teammates here. Their support and contribution has enabled this project to be successfully completed, and allowing our website to finally stand out from the class.

In the end, With the sincerest thankfulness, I would like to extend my highest respect to each teacher and tutor. Like Andrew mentioned in the end of the final presentation of the projects, each of us started from zero, and now we managed to step into the door of the programming thanks to you. We would never achieve today’s goal without your self-giving support and patient guidance. The course of my career is literally changed, and this will definitely be the most unforgettable period in my life.

Please again, accept the assurances of my highest consideration.

8th,June,2018.

Yang Tianyi

tyan227 644294325