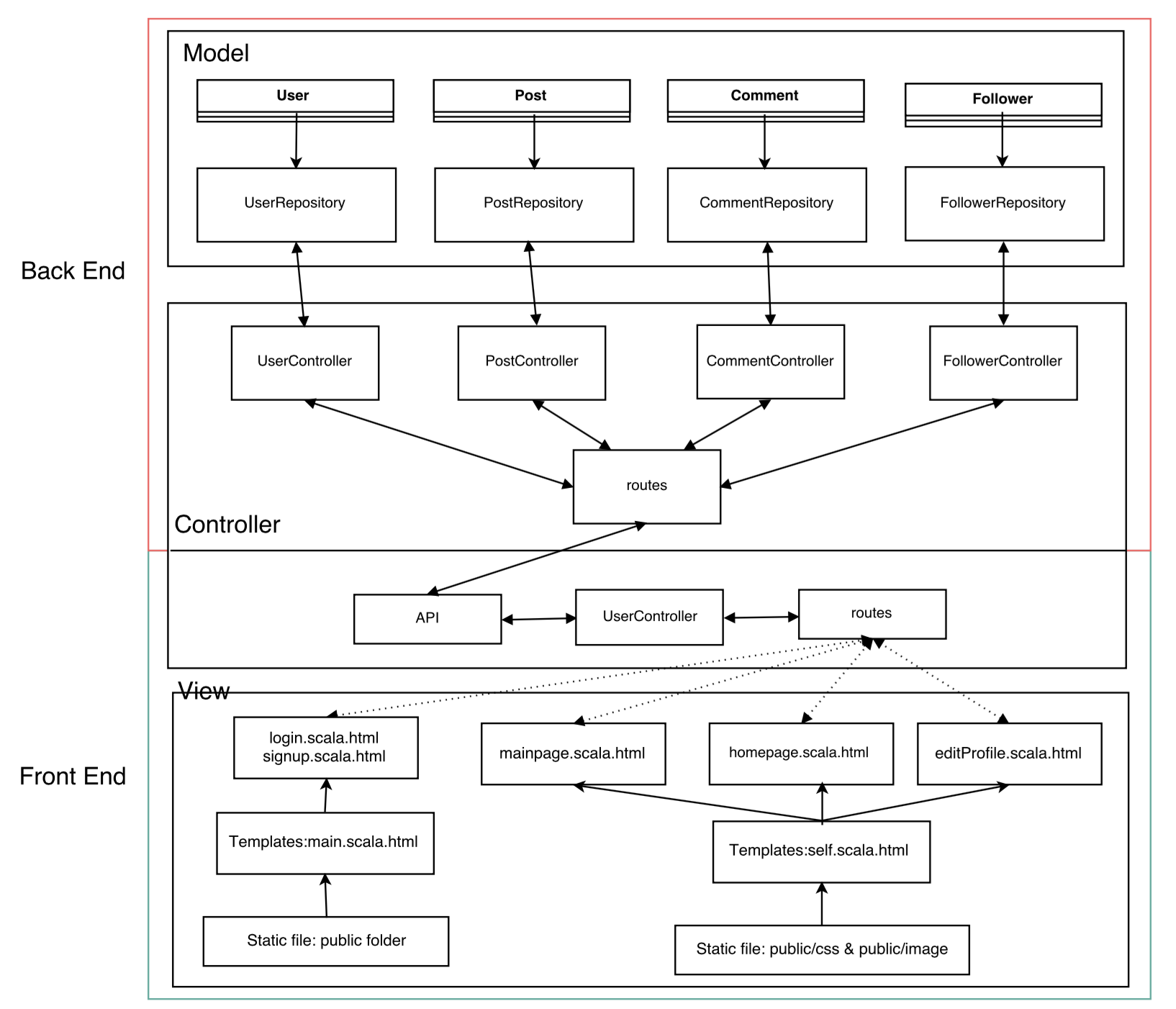
Design documents

**System design**

First we are going to give an overview of the system design of our project. Then we will introduce the front end and back end system design respectively. After that, brief introduction of our database schema will be presented.



**Front end**

**Back end**

For back end system design, there are four main models, which are User, Follower, Post and Comment. Each of them corresponds to a table in the MySQL database and has a repository and controller respectively.

At first, here are the screenshots of our design drafts of these models.

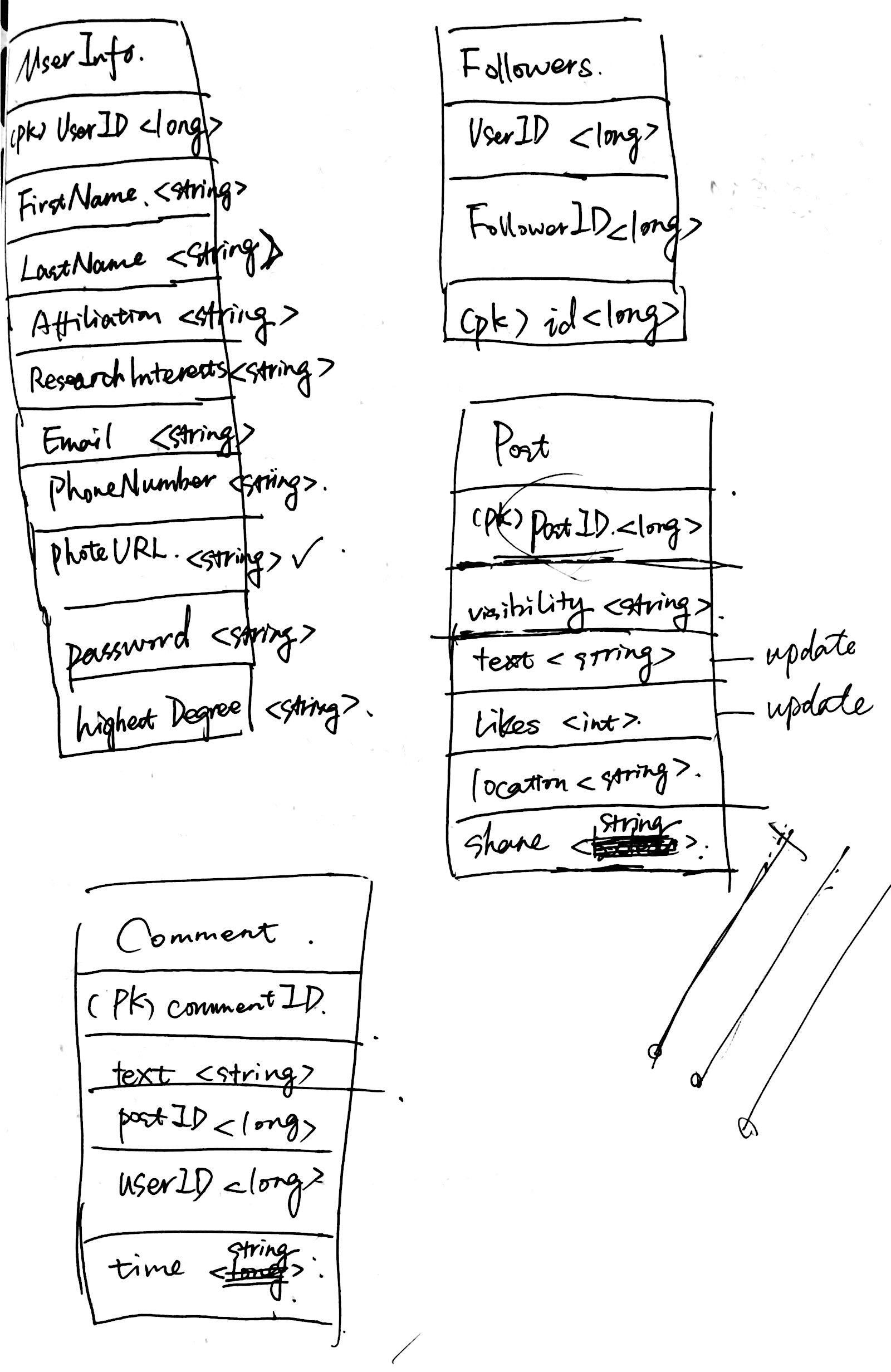


Fig. 1 Design draft of database schema

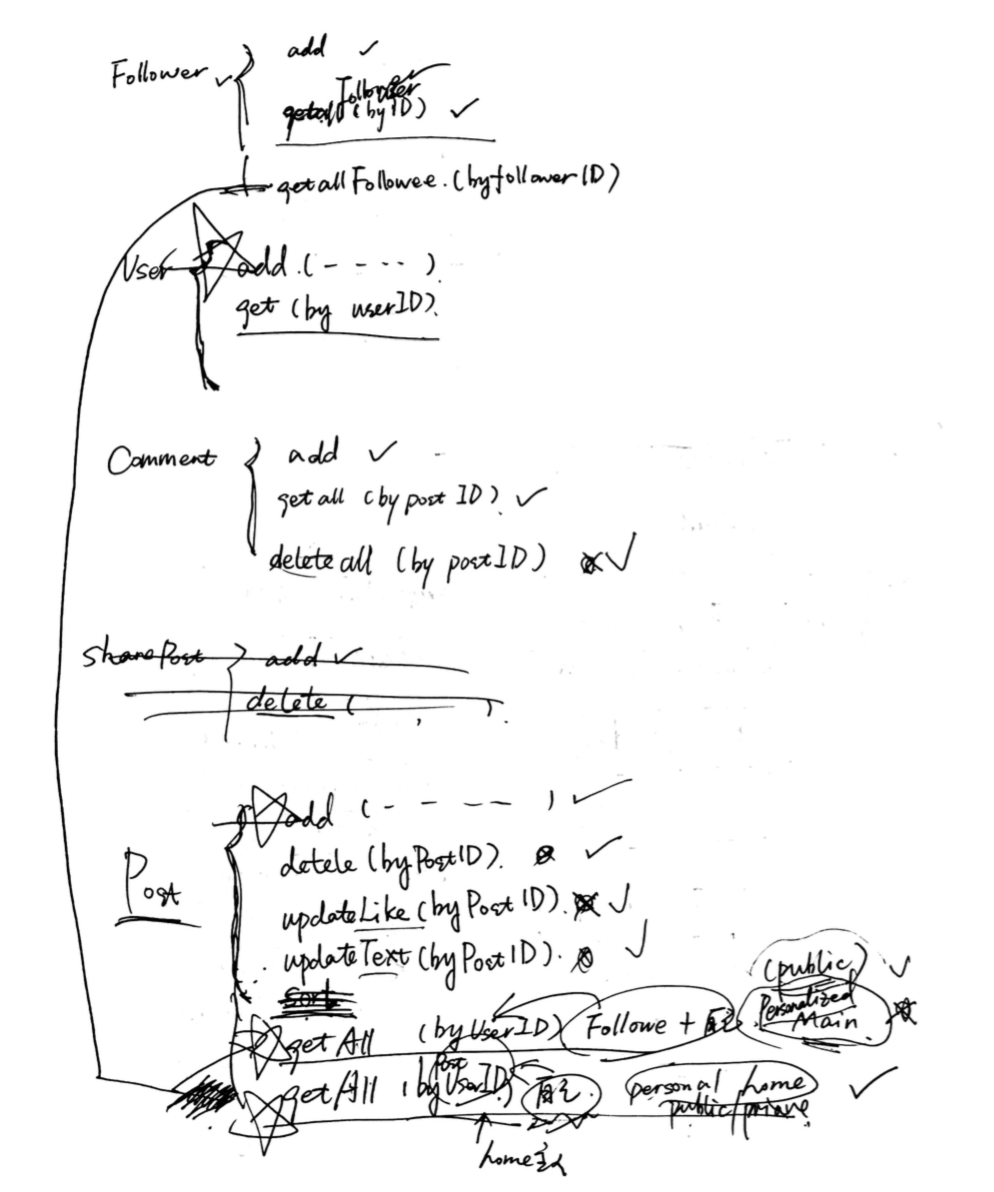


Fig. 2 Design draft of back end implementation methods

Then, here are some detail explanations and screenshots of each class.

1. User

For user model, its primary key is the automatic generated id. It has many fields to store the user-related information such as email, password, first name, last name and so on. When there are some operations in the front end that need to check or edit user’s information, such as log in, corresponding methods in User controller will be called. Here are the screenshots of User class and User controller.

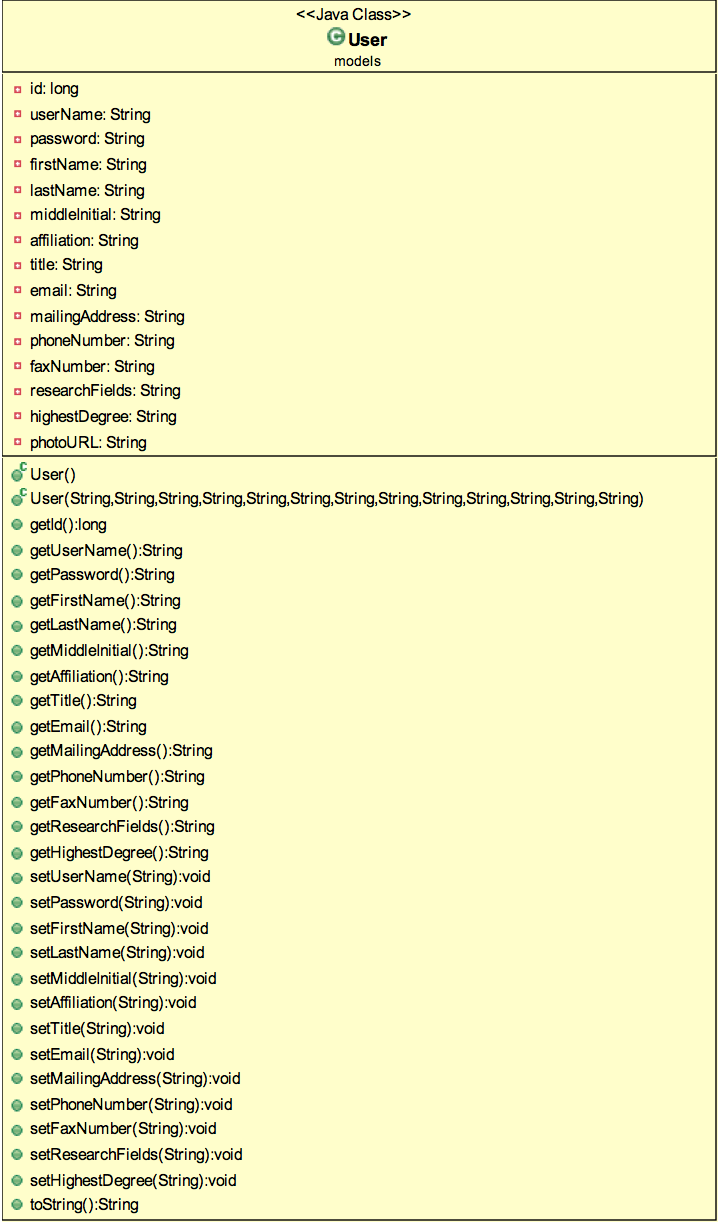


Fig. 3 User class

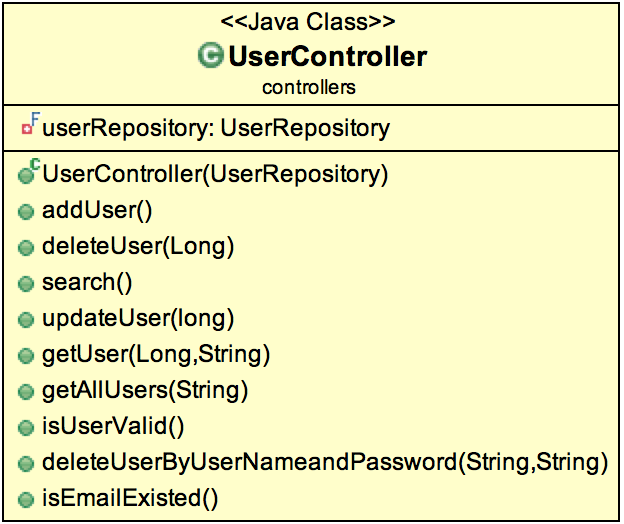


Fig. 4 User controller

1. Follower

For follower model, it only has three fields. It uses the automatic generated id as its primary key and has another two fields, user Id and follower Id. This model is used to record all the follower and followee relationship. When there are some operations in the front end that need to check or change user’s follower/followee information, such as following a new person, corresponding methods in Follower controller will be called. Besides, it relates with User model closely by using the id fields. Here are the screenshots of Follower class and Follower controller.

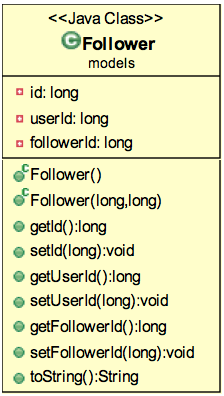


Fig. 5 Follower class

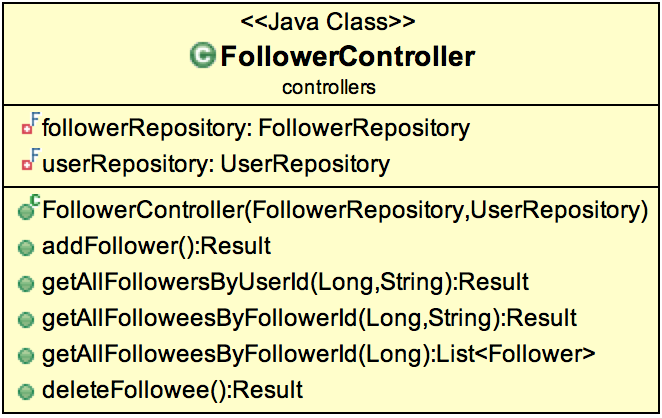


Fig. 6 Follower controller

1. Post

As for post model, it also has an automatic generated id as its primary key. Considering all the post-related operations, like publishing public/private posts, edit/like/delete/share posts and share location, post model has ten fields altogether including id. Each of these fields is used to store post-related operation information. When there are some operations in the front end that need to change post information, such as publishing a new post, corresponding methods in Post controller will be called. Here are the screenshots of Post class and Post controller.

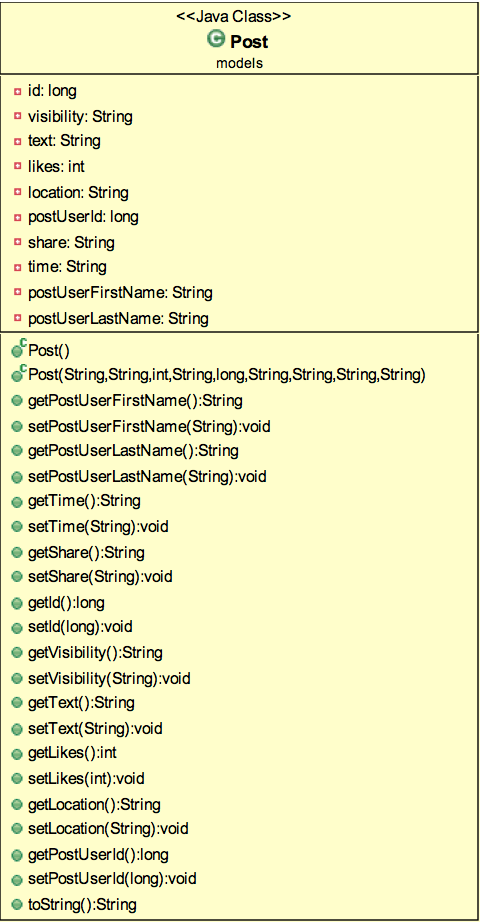


Fig. 7 Post class

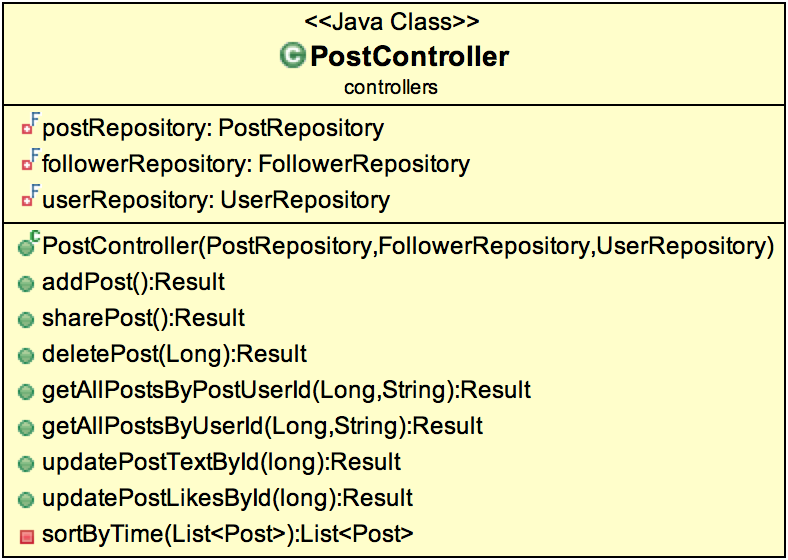


Fig. 8 Post controller

1. Comment

At last, for comment model, like others, it sets an automatic generated id as its primary key. Except for that, it has a post Id field to recognize which post it belongs to, user information fields to specify who adds this comment and a time field to record when it is added. Comment class is related to Post class closely using the field post Id. When there are some operations in the front end that need to check or add comments, corresponding methods in Comment controller will be called. Here are the screenshots of Comment class and Comment controller.

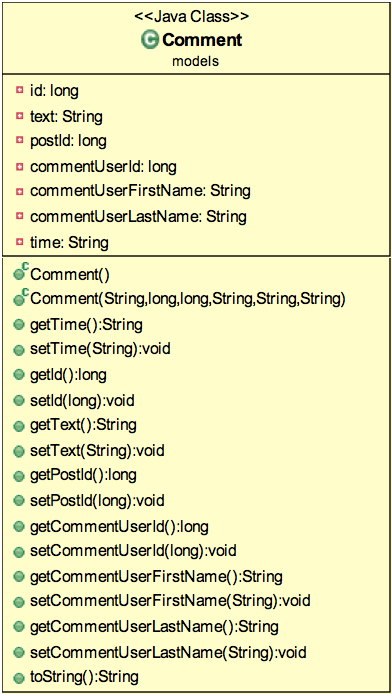


Fig. 9 Comment class

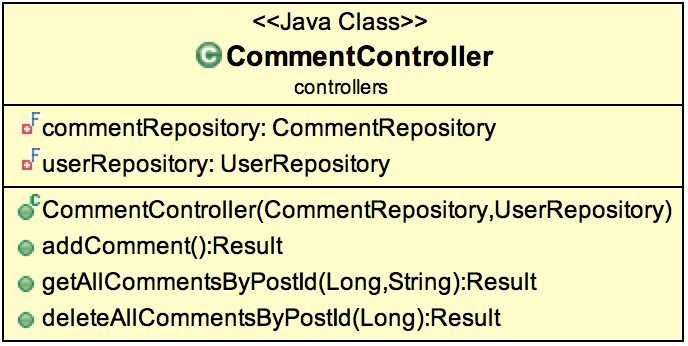


Fig. 10 Comment controller

Finally, following is the screenshot of our final database schema. From this picture, we can see the detail information of each table and the relationship between them.

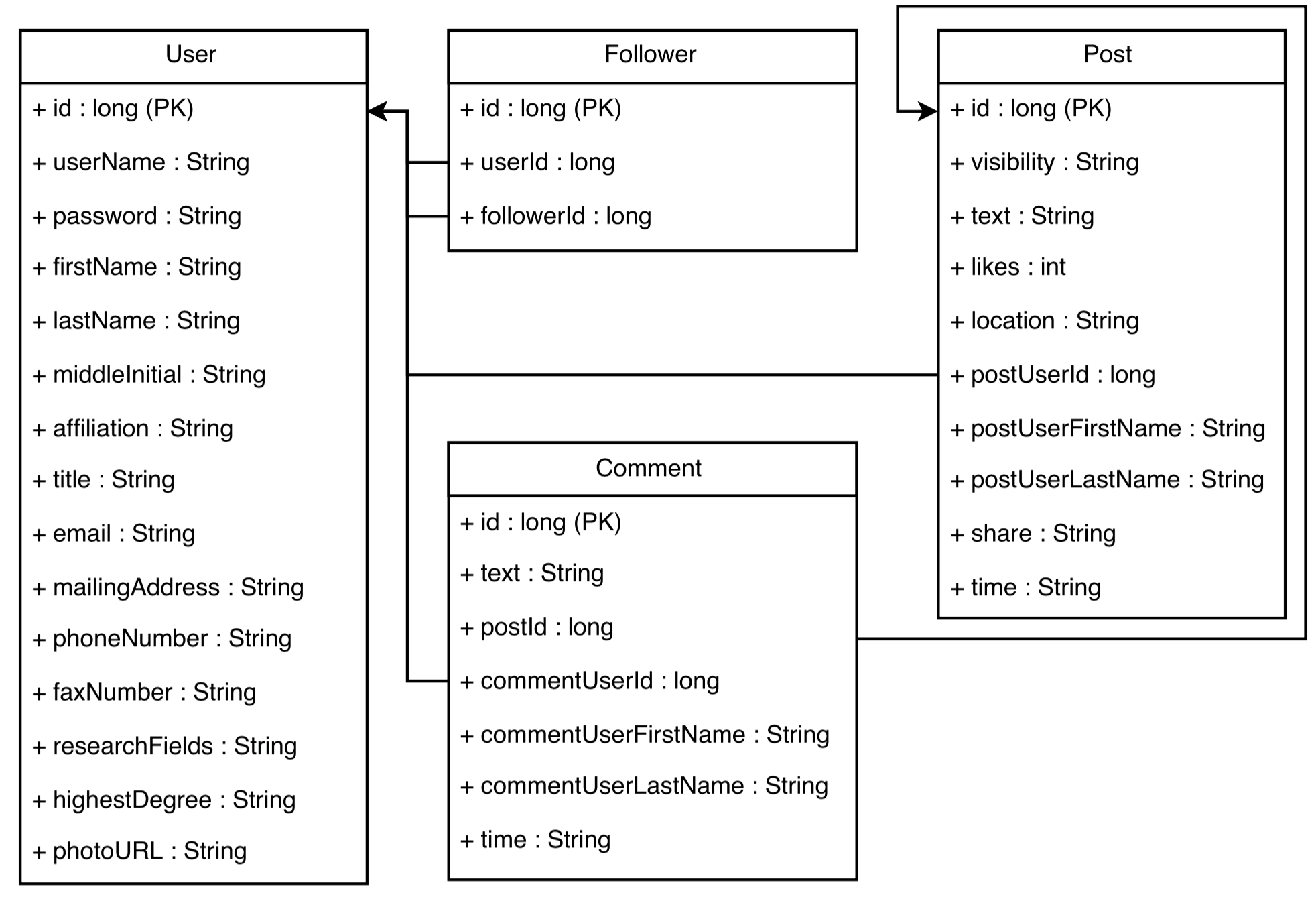


Fig. 11 Final database schema