Explanation

It’s been a busy week. As I ran out of time, I haven’t got a chance to write test cases and thorough comments in the code, though I tried my best to make the code self-explanatory.

Site address

http://54.86.117.51/

Available Users

Name: jack@abc.xyz
Pass: jj
Roles: user, admin

Name: thousandEyes@abc.xyz
Pass: 123
Roles: user

Pages

Admin page: http://54.86.117.51/admin, where you can see all the users, and add new users.

Index: http://54.86.117.51/, where you can

1. See who you are
2. See who you’re following
3. See who are following you
4. See all posts from yourself and people you’re following
5. Follow/Unfollow others
6. Post new posts

Ajax APIs (all require login, all return JSON)

GET /user/allUsers (get all users)

POST /admin/create?username&name&password&roles (create new user)

GET /post/allPosts (get all posts can be seen by current user)
POST /post/publish?content (publish a new post for current user)

GET /follow/getFollowers?id (get all followers for current user)

GET /follow/getFollowees?id (get all followees for current user)
POST /follow/follow?id(user to be followed) (follow a user for current user)
POST /follow/unfollow?id(user to be unfollowed) (unfollow a user for current user)

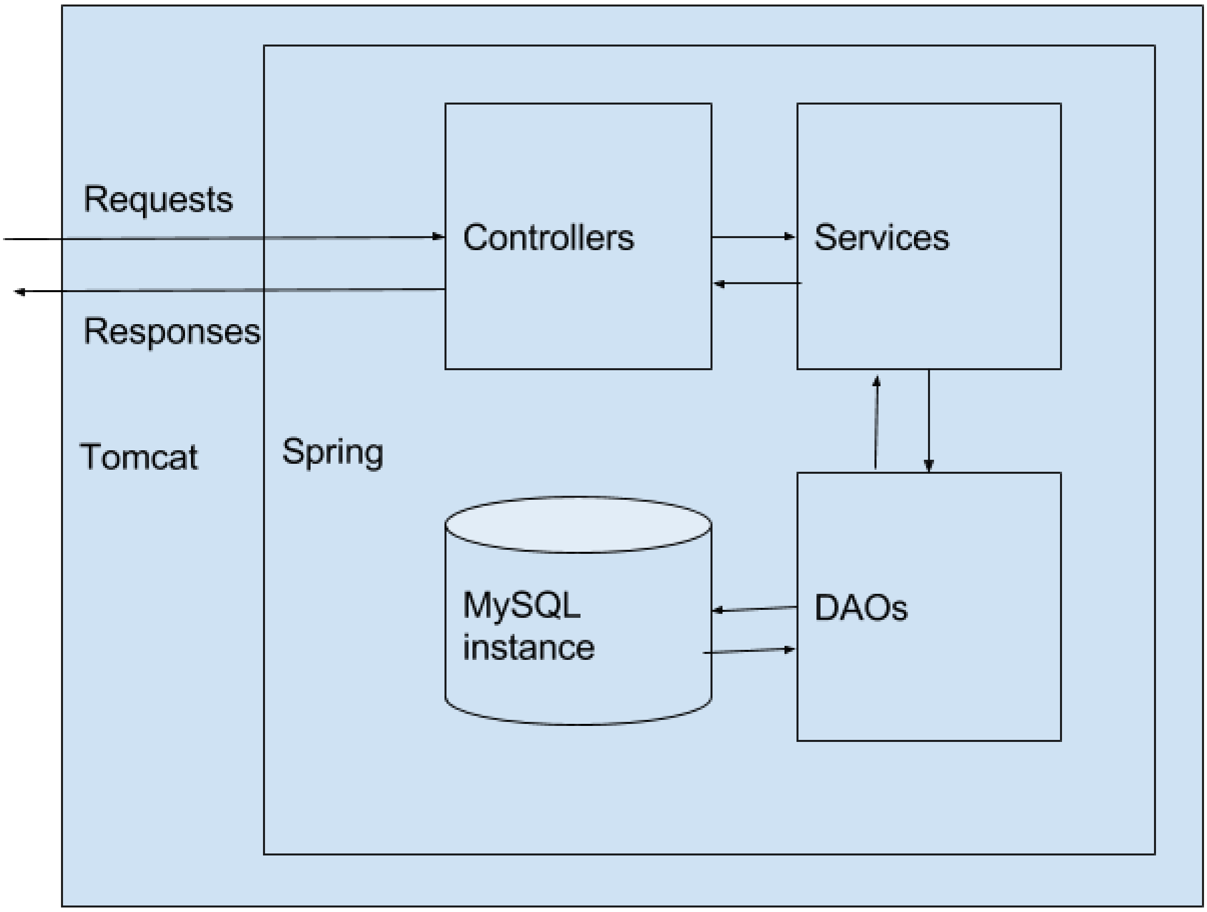
Spring features used

Basic features, MVC, NamedParameterJdbcTemplate, Spring Security, Spring transaction management

MySQL features used

Basic features, full-text index

Software Architecture



Github

https://github.com/magicyuli/Twidder