

PROJECT PART 4

51: IN CLASS COMPETITION BOARD

April 12, 2018

Bu Sun Kim
Kumar Bhargav Srinivasan

1: PROJECT SUMMARY

We want to create where faculty register and can post coding competitions for students to participate in. Teachers should be able to create competitions, evaluate the submissions and view the submissions and results. Students should be able to register, participate in the competition by submitting code and view the results. Faculty should be able to end the competition. An admin user is responsible for granting permissions for students and faculty.

2: PREVIOUS CLASS DIAGRAM (FIXED)

Please see page 5

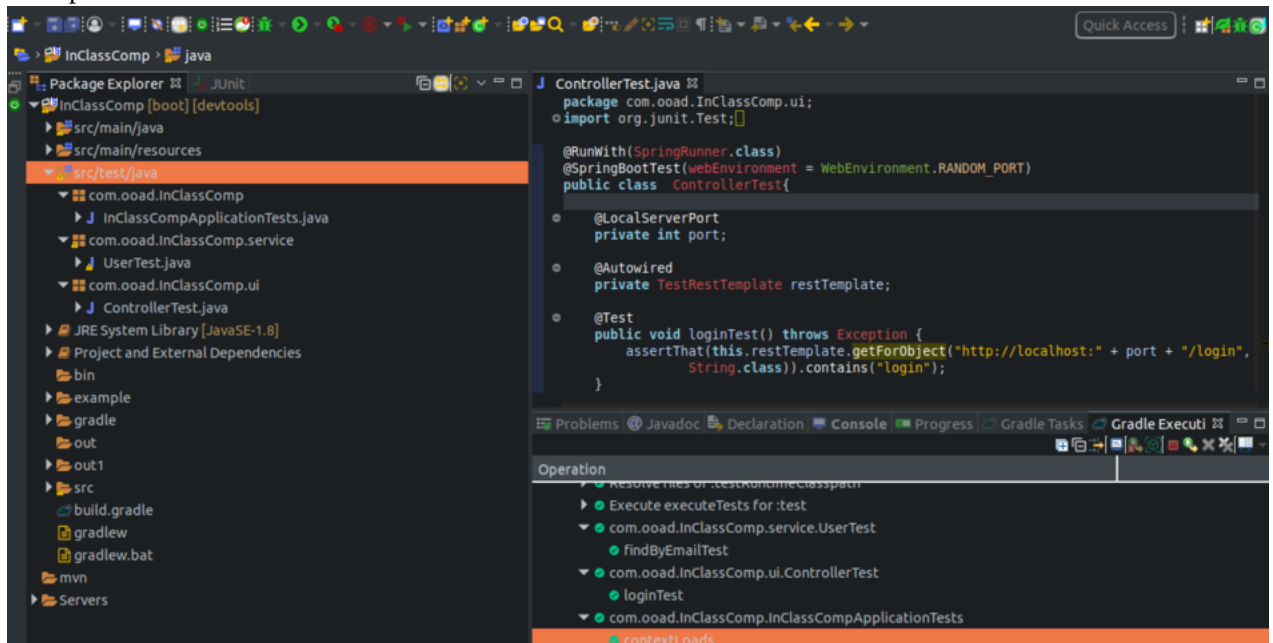
3: COMPLETED CLASS DIAGRAM

Please see page 6

4. SUMMARY

- 1) Implemented Design patterns for common problems (Facade and Factory pattern)
- 2) Implemented couple of UI pages to consume API created
- 3) Unit test cases for DAO and Controllers
- 4) Refactor by moving out common code to Utils/Service

Example testcase:



5. BREAKDOWN

Bu Sun:

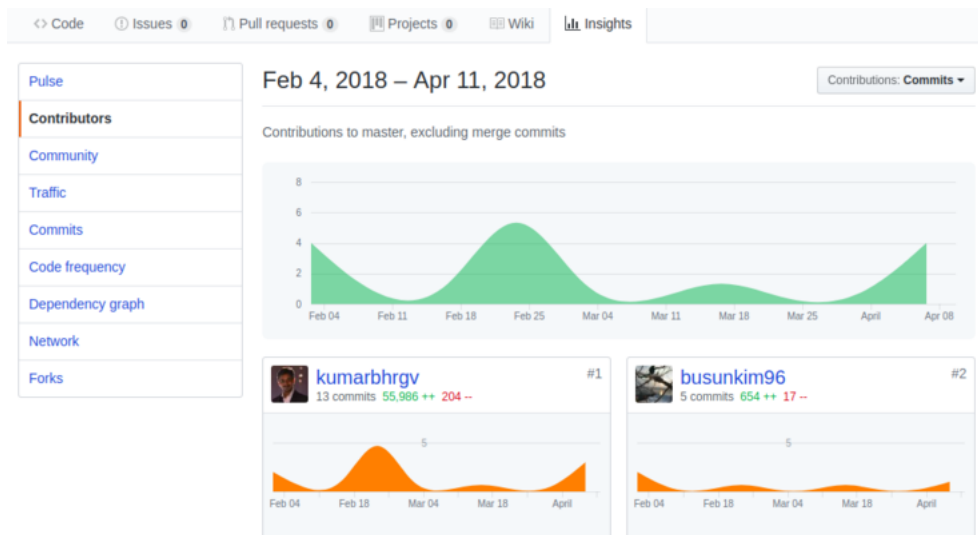
- 1) Implemented view competition, create competition, and add users UI flow
- 2) Implementation for consuming competition API

Kumar Bhargav

- 1) Design pattern implementation
- 2) Documentation
- 3) Couple of test cases

We had couple of meetings and conversations to sync up.

6. GITHUB GRAPH



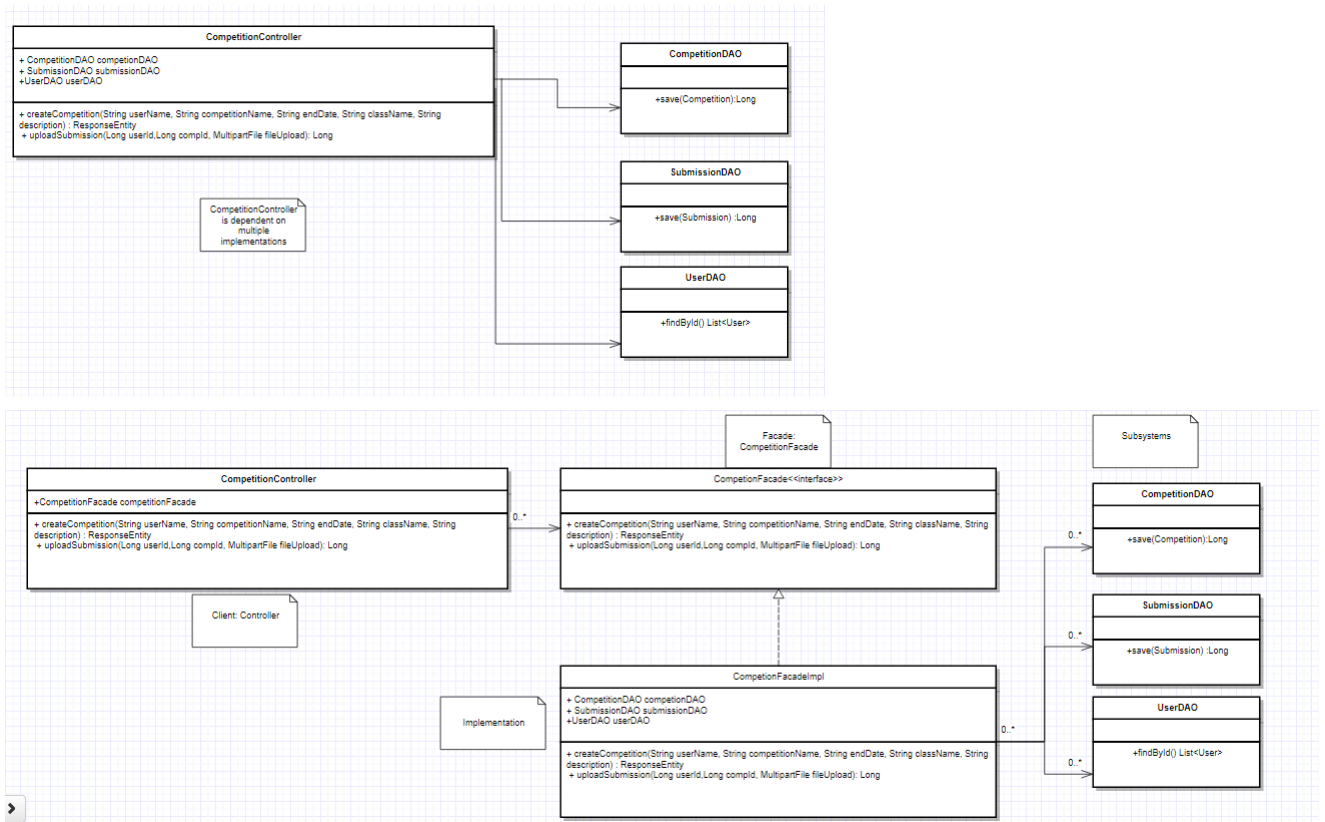
7. ESTIMATE REMAINING EFFORT

- 1) Implement Session management, Security, Documentation to it. 6 hours.
- 2) Continue to update our Class Diagrams.
- 3) Continue to work on user Interface to consume API's written - 6 hours.

8. DESIGN PATTERN

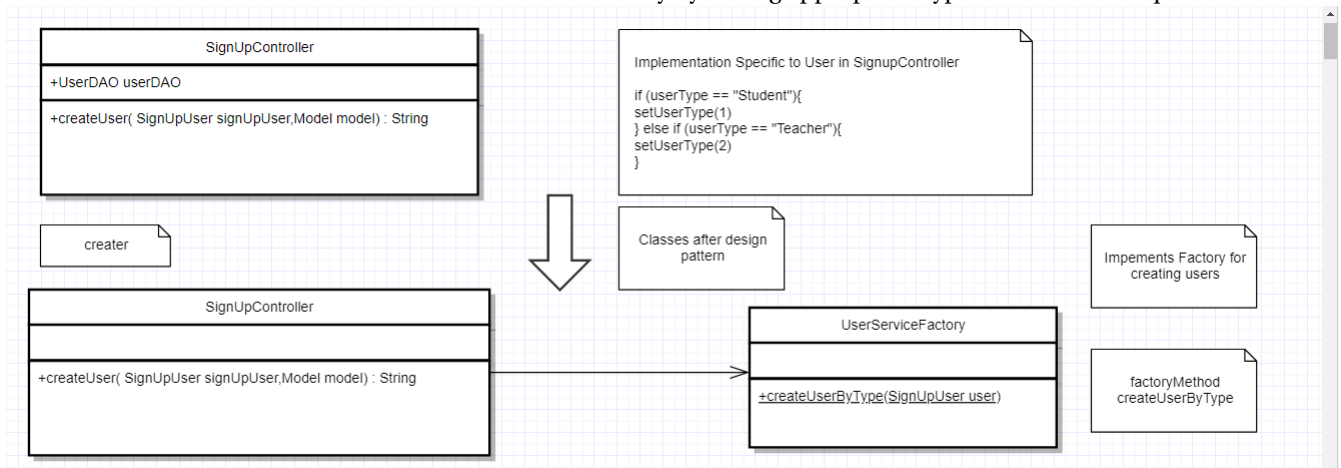
Facade Pattern: Act as an Interface between client controller and Data Access Objects. We will have flexibility in changing the underlying Database implementation of the application without any change in the controller code, thereby decreasing coupling.

Following are the diagram of classes which were involved:



Factory Pattern: Create User Objects using factory

Unified interface to create users such as Student and Faculty by setting appropriate type based on the request



9.NEXT ITERATION

- Continue Develop test cases.
- Continue work on frontend. Consume API endpoints instead of using mock variables.

-start applying refactoring principles.

