Second Iteration Demo

Team: LongNameWillBeRemembered

Chengqi Dai (cd3046), Yiqian Wang (yw3225), Wenbo Song (ws2505), Zhongkai Sun (zs2341)

Fall 2018 - COMS W4156 Advance Software Engineering - Prof. Gail Kaiser

1. The date and time at which you already completed this demo, and briefly describe any challenges that arose before or during the demo.

We completed this demo after class on Dec 4th. The challenge during the demo is that when we create a online game room, the program sometimes crashed. Also, there are something we need to add in our program such as the username and password label, the notifications that tells the user about his turn and the quit button in the game.

2. The specific use cases that were demonstrated, highlighting any changes since the First Iteration Demo.

We added the test cases for loop initiation, continuation and termination:

For the client part, since we have a nested loop in our program, we test the inside loop first to test cases for loop initiation, continuation and termination. We set up a number of valid inputs first. Then, the inside loop will check if there is five row in a line by input a piece value, x & y position and direction from outer loop. We test the inside loop first for different inputs by our test case to test if it could be terminated and give a correct return value. Then we check the outer loop to test if the outer loop works correctly.

For the server part, we test users name and password when login, and test the room id when players enter create or join a room. When the game starts, we test the socket to make sure the data are correctly transmitted.

3. The specific CI mechanisms that were shown during the demo, including which technology you used.

Post-commit CI: The detail for post-commit test is inside the file .travis.yml including language, version, install and requirements. The post-commit run both python testcase for client and java test cases for server.

client/test.py -- the post-commit file includes multiple boundary conditions and potential faults test and loop tests.

server/src/test/java/com/gomoku/*.java -- we use maven to build our server, so, for the post-commit of server, we run the maven test. We test the game logic on server, game socket and http request controllers (besides the controllers we have in the first iteration, we added room controller).

Travis CI could build our post-commit test and report the job log result.

We use Coveralls to add a coverage tool to the post-commit CI process, the branch coverage is 95%.

4. A link to the github repository where your entire codebase resides. Tag the revisions that were shown in the demo.

Link: https://github.com/OliviaWYQ/Gomoku-Desktop   Tags : V2.0
Travis-CI: https://travis-ci.org/OliviaWYQ/Gomoku-Desktop
Coveralls: https://coveralls.io/github/OliviaWYQ/Gomoku-Desktop