Northeastern University
CS 5500 – Managing Software Development
Fall, 2018, Jump, Katz, Weintraub

Sprint 2 Review

Team: **111**

TA: **Ruchitha**

Review Date: **17 November,2018**

Base Expectations

1. Getting the legacy code base running.
   Comments: Working.

2. Adding a notion of user and groups of users to the system. The entities should be persistent. Basic User/Group management functionalities should be provided (i.e. Create, Read, Update, Delete)

Comments: Achieved persistence using LDAP.

Note: LDAP is mostly used for user validation may not be a good fit for message persistence.

3. Directing messages to individuals and groups. Messages that are replies to messages sent to a group go to all other members of the group.
   Comments: working

4. The project is using at least version 1.8 of Java.
   Comments: yes

5. The system is packaged as a standalone system (it should not require being run in an IDE).
   Comments: yes

6. The system is deployed to a cloud environment such as Amazon AWS, Azure or Mass OpenCloud. You are encouraged to use a free service. You must provide administrator access to the project executives.
   Comments: Continuous deployment not done

7. Work is being managed in Jira.

Comments: Jira is not reflecting all the work completed. Need to work on JIRA to manage your tasks.

Stretches

8. (medium) Adding a login with username/password with password storage being encrypted.
   Comments: +3 for login feature and encrypting password

9. (medium): Message persistence to users
   Comments: +2.5 for user message persistence.

10. (medium): Message persistence to groups
    Comments: +2.5 for group message persistence.

11. (large) MIME support to messages:
    Comments: Not done

12. (small): team is using smart commits in git.
    Comments: working

13. (small): Jenkins should inform the team of failure either in Slack or email.
    Comments: working

14. (small): Github should inform the team of PRs via slack or email.
    Comments: Working

+2 -> extra credit to accommodate smaller team size

Team Score: 95

Additional notes:

1. Good effort in designing database. LDAP is targeted for read-mostly access, which makes it ideal for user validation. However, it is likely not a good fit for message persistence. Can make use of RDBMS to make it more efficient as it improves the message retrieval operation by making use of indexing internally while querying the table.

2. Store data on server instead of having it run on local server.