Final Presentation

Farha Jawed
Pratik Devikar
Yang Xia

Team Member Introduction

- Team 111
  - Farha Jawed
  - Pratik Devikar
  - Yang Xia

System Functionalities

Did the team achieve a sufficient amount of functionality?

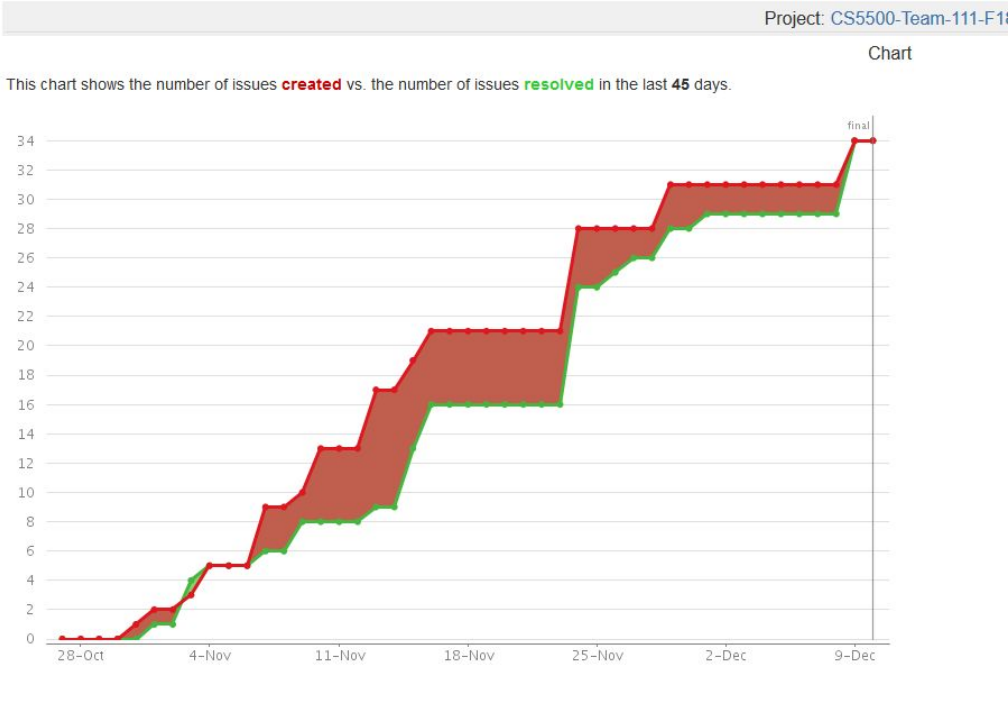
Yes. The system is working properly.

All the basic functionalities that a simple messaging application should have are being implemented.

Ex: Sending messages to other users and groups, storing messages for later retrieval and sending messages to offline users.

System Functionalities

Fig: Issues created vs resolved



System Functionalities

Did the team achieve what it set out to do?

Yes. We did complete all the sprint expectations.

Getting the legacy code running which contains Prattle and ClientRunnable, Crud functionality for users and groups, sending messages to and from users and groups, delivering messages to offline users and providing a service for an agency to view the messages. Our system also provides the users with parental control.

System Functionalities

Although being a member of 3 people, we weren’t able to complete some of the stretches such as adding MIME(Multi-purpose Internet Mail Extension) type to messages and using Apache JMeter to test the system for stress testing and Performance testing.

System Functionalities

Is it useful for a client?

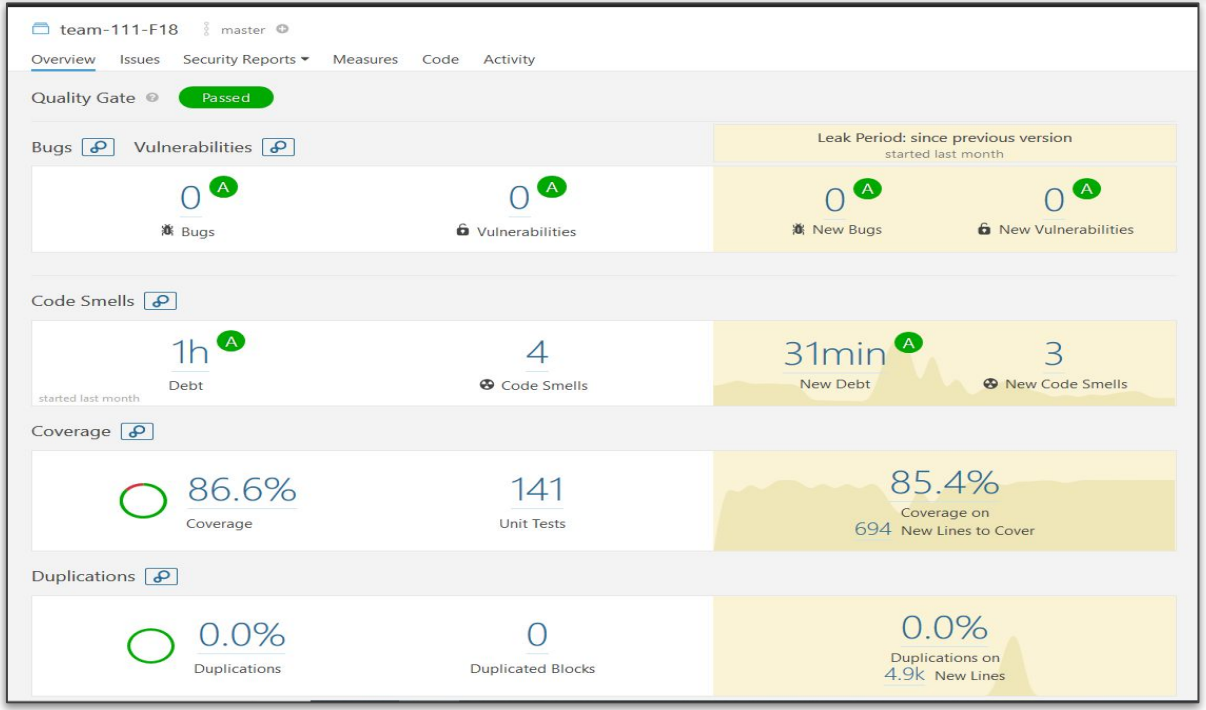
Yes, most of the basic back-end system functionalities have been implemented. But a user also desires of a good GUI for any application. We didn’t do the chatter/user interface. So in the future if someone develops the chatter side of the project, it will be ready for deployment.

Job Quality

- Did the team present evidence that it did a quality job?

- Did the team improve its performance over time? (And is this claim supported with evidence?)

Job Quality Contd.



SonarQube Report

- 86.6% code Coverage
- 141 unit tests
- No bugs and vulnerabilities
- 0% code duplication

Job Quality Contd.

Performance over time

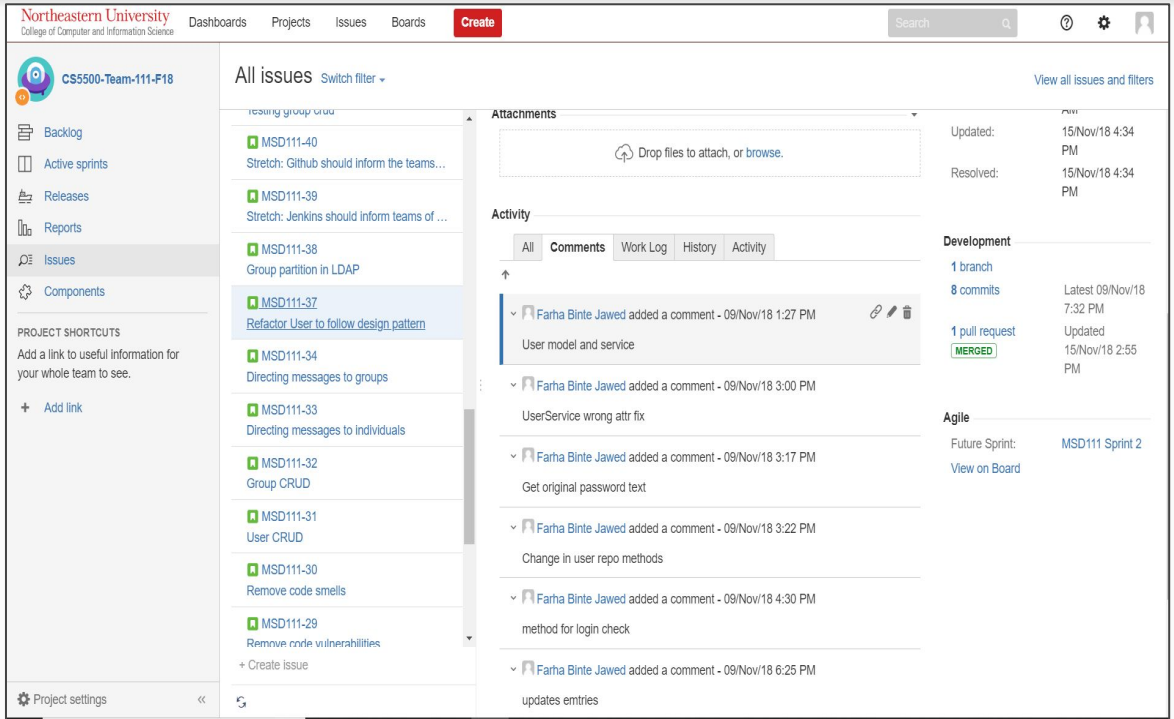
- Sprint 1: barely met the minimum requirement
  - Wrote test cases for Prattle

Job Quality Contd.

Performance over time

- Sprint 2 and 3: Could do more stretches
  - Worked efficiently by prioritizing tasks on Jira
  - Distributed works evenly among team members

Job Quality Contd.

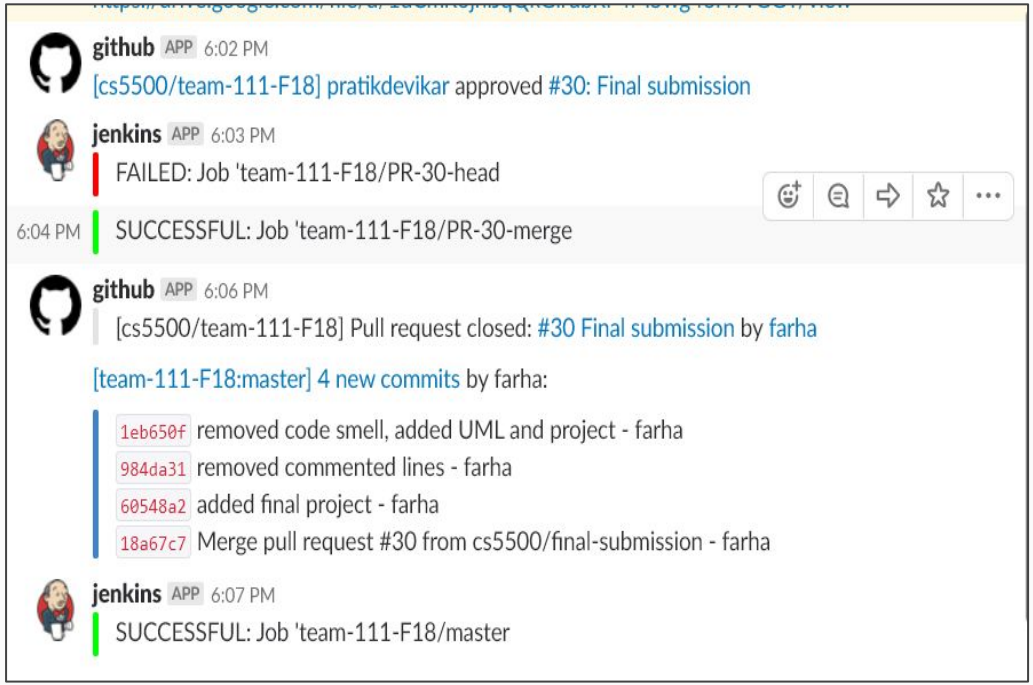


Performance over time

Sprint 2:

Started using smart commits on Git

Job Quality Contd.



Performance over time

Sprint 2:

Integrated Jenkins and Github with Slack

Job Quality Contd.

Performance over time

- Team met more frequently

- Incorporated feedback from sprint reviews

Process and Team Work

Did the team work as a team?

Yes!!!!!!!

The fact that we as a 3 member team is able to finish the project is the best proof.

Process and Team Work

Did the team use process well or was the project mis/mal-organized?

Show Jira.

Assigned tickets, update stories, and smart commits.

Slack webhook

Email notifications

Process and Team Work

Was the team able to automate the build, test, and promote process?

Jenkins/SonarQube.

Technology Transfer

Is is the system in shape to be handed over to the client?

1. Client side needs more work
2. Still had some issues with LDAP server
3. Need to modify the way we persist messages

Future works:

1. Fixed the issues that mentioned above
2. Writing test cases using a real server in addition to Mockito
3. Provide user interface