CS 5500
Project Presentation

Team 213

System Functionality

- User functions
  - Login / Register new user
  - Check/update profile
  - Can set do not disturb
  - Follow/ unfollow others
  - Subscribe/ Post to thread
  - Retrieve a chat
  - Reply to a message
  - Forward a message
  - Send secret message
  - Check if a user exists
  - Recall a message if not viewed
  - Get messages between dates
  - Can send a file to another user
  - Get a reply chain of messages
  - Check last seen of a user

- Group functions
  - Users can create/ delete groups
  - Users can send a message to a group
  - Users can add others to a group
  - Set group restriction to high or low
  - Check who is in the group
  - Multiple group admins and ability to grant others admin privileges

System Functionality

- CALEA compliance functions
  - CIA user can wiretap / untap a user
  - CIA will get a live notification of any message that is sent or received by a wiretapped user
  - CIA can retrieve all the communications (send and received) of a wiretapped user since the wiretap was set
  - CIA can see a list of users who have been wiretapped
- System functions
  - Password hashing
  - End to end message encryption
  - Deployed on AWS instance
  - Caching of groups and chat ids for faster retrieval

System Functionality

- Client requests
  - Groups with moderators
  - Sending individual and group messages
  - Reply to messages
  - Security, CALEA & Encryption
  - Push notifications
  - Recalling messages
- Deliverables
  - Focused on high priority (must have) features
  - We were able to implement nearly all must have features
  - Some features were not possible due to conflicts with other features
  - Delivered extra features that went over and above client expectations

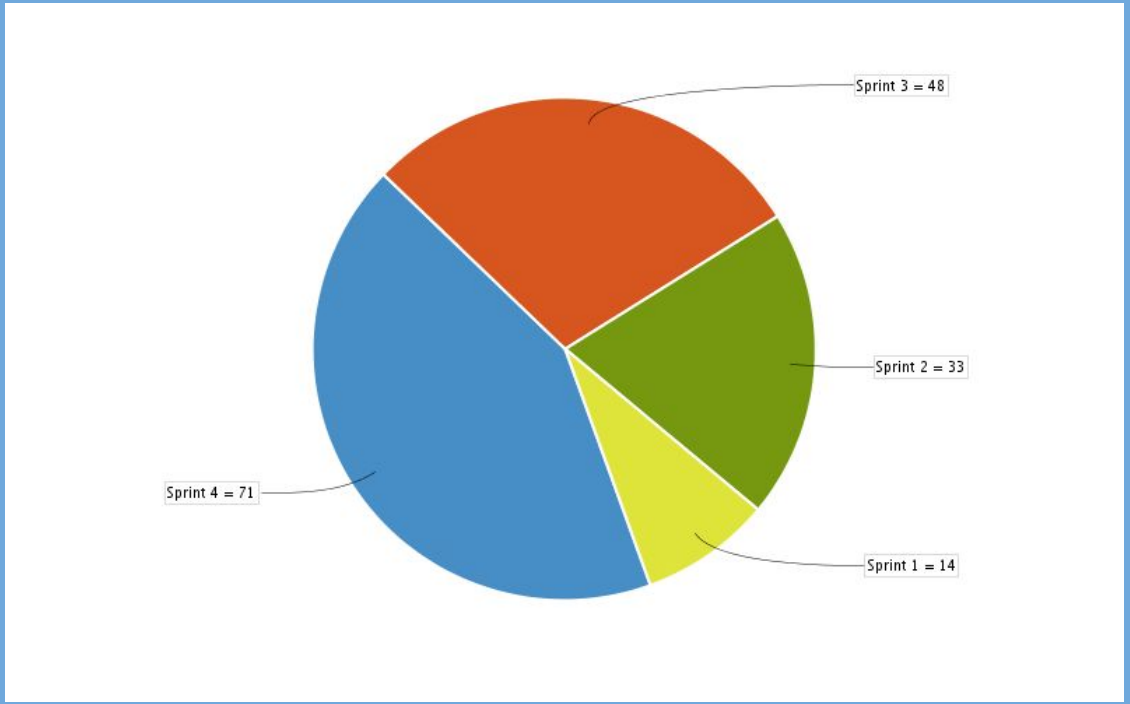
Job Quality

- Focus on writing clean code in order to improve readability
- Ensured there were jira tickets assigned to track the tasks by incorporating smart commits
- Wrote well documented code
- Each deliverable was completed and presented on time
- Under-promised and over-delivered
- As shown in the pie chart in the next slide, our efficiency as a team increased overtime. We were able to implement and test many more features efficiently, in the same amount of time, in the latter sprints.

Job Quality

Data Table

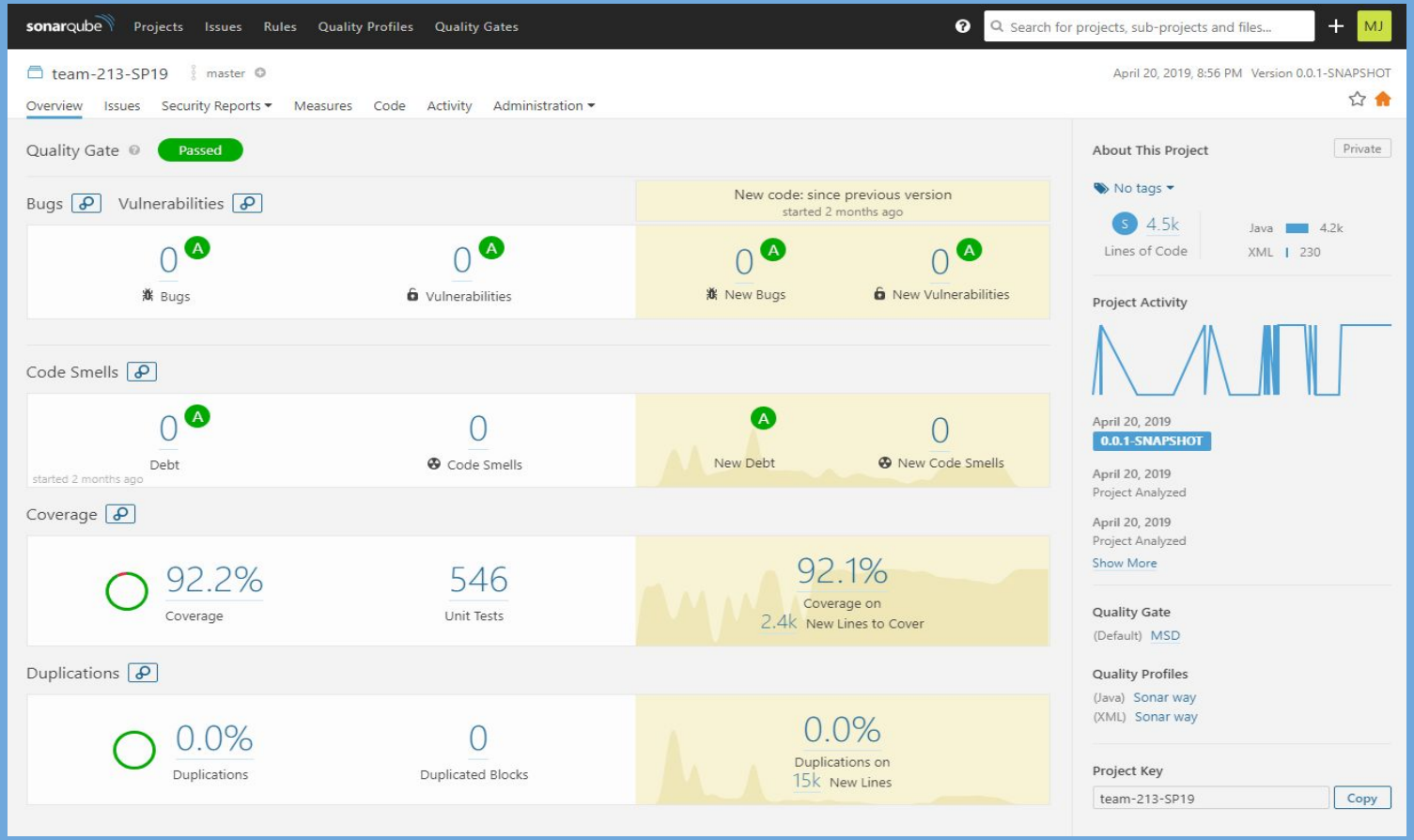
	Issues	%
Sprint 4	71	44%
Sprint 3	48	29%
Sprint 2	33	20%
Sprint 1	14	8%



Job Quality

- Maintained high code coverage with thorough rigorous tests
- Provided a test suite that serves as a regression pack to protect existing tests against additional features added in the future
- Focused on both functional testing as well as structural testing in order to ensure that the system features are robust.

Job Quality


sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Search for projects, sub-projects and files...
MJ
team-213-SP19
master
April 20, 2019, 8:56 PM Version 0.0.1-SNAPSHOT
Overview
Issues
Security Reports
Measures
Code
Activity
Administration
Quality Gate
Passed
Bugs
Vulnerabilities
New code: since previous version
started 2 months ago
0 Bugs
0 Vulnerabilities
0 New Bugs
0 New Vulnerabilities
Code Smells
0 Debt
started 2 months ago
0 Code Smells
New Debt
0 New Code Smells
Coverage
92.2% Coverage
546 Unit Tests
92.1% Coverage on 2.4k New Lines to Cover
Duplications
0.0% Duplications
0 Duplicated Blocks
0.0% Duplications on 15k New Lines
About This Project
Private
No tags
4.5k Lines of Code
Java 4.2k
XML 230
Project Activity
April 20, 2019
0.0.1-SNAPSHOT
April 20, 2019
Project Analyzed
April 20, 2019
Project Analyzed
Show More
Quality Gate
(Default) MSD
Quality Profiles
(Java) Sonar way
(XML) Sonar way
Project Key
team-213-SP19
Copy

Process and Teamwork

- We worked well as a team with all the work being equally distributed
- 7-8 meetings per sprint on average
- Met in person
  - Started with discussion of what needed to be finished short term vs. long term
  - Practiced pair programming
  - Continuous code review
- Communication was consistent through Slack.
- Improvement as sprints went on
- Followed the Agile process wherein daily updates and action items were given to the scrum master by each of the team members
- Mandated feature-test methodology

Process and Teamwork

- The team was very organised and the flow of the project was well structured.
- The deliverables were divided well between the sprints.
- Whenever we came across any issues, we would discuss it as a team and then the Scrum Master would talk to the client about it in order to ensure that the team overcomes it.
- Team used the automation for build and test by incorporating Git, SonarQube (for metric evaluation), Jenkins(for build) and JIRA (for which we used smart commits)

Technology Transfer

- The product built can be delivered to the client
- A UI would greatly help with user experience and ease of use
  - Buttons to replace wordy commands
- Adding a feature to dictate what the user says into a typed message by using Natural Language Processing
- Including translation of messages into different languages

Thank You!