

Final Presentation

CS 5500 Foundations of Software Engineering

Jainam Sheth
Devanshi Ganatra
Purvil Bambharolia
Dipen Patel

System Functionalities

Basic Functionalities

- User Functions
- Message Functions
- Group Functions
- Government Functions

Completed all 88 issues

- 45/45 must have requirements were finished.
- 7/15 good to have requirements were finished.

Achievements

Goal : Build a Java supported web socket backend for the chat system
Sprint Goals:

- S1 : Integrate prattle into build system.
- S2 : Integrate Database with the system and implement features for User to User messages.
- S3: Feature of messaging in a Group and other Message features like Media upload and message forwarding.
- S4: Government Functions and features such as Polls and Threads in Group messaging.

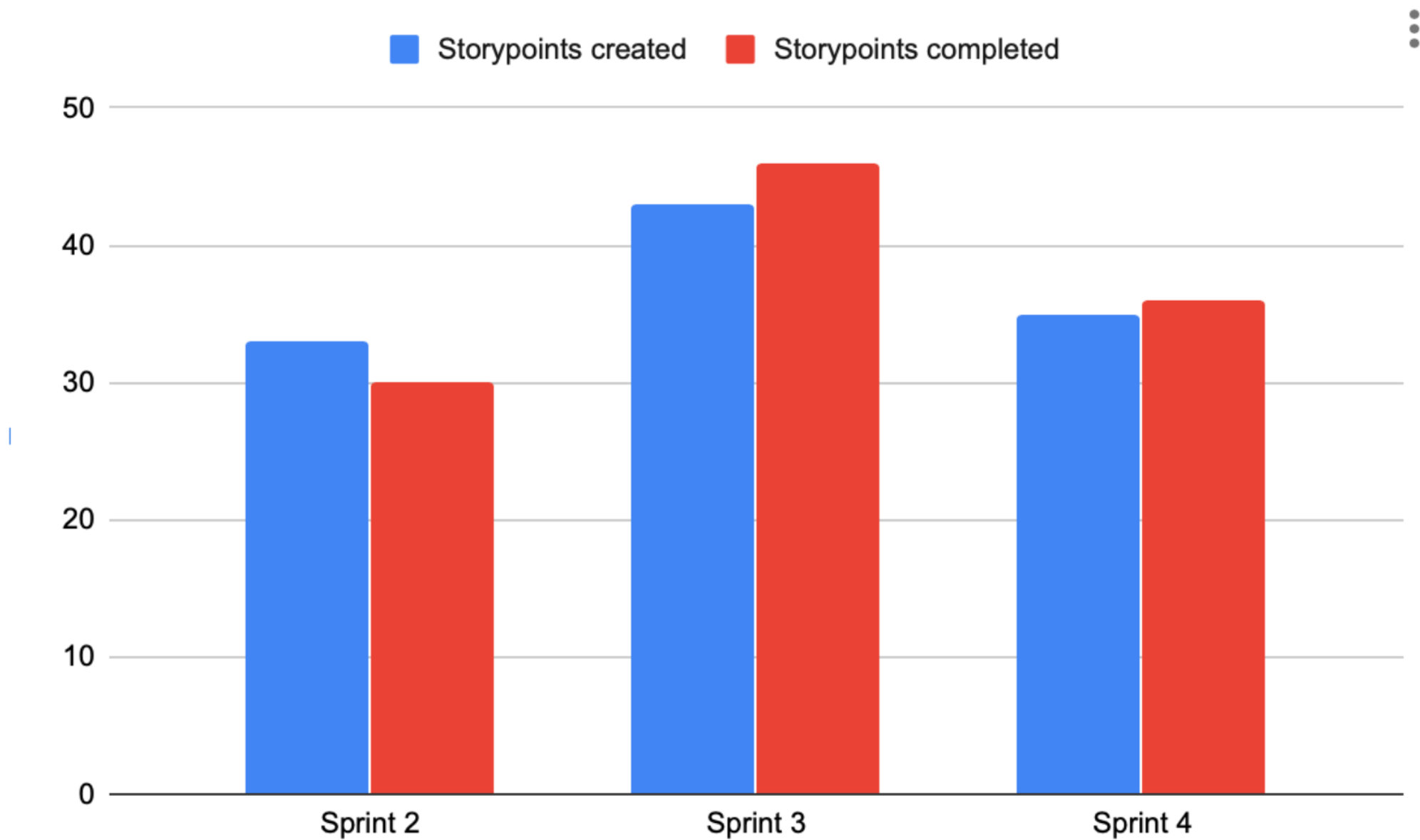


Fig. Sprint story points

Usefulness to Client

- Provides the client to have secure messaging within the team or the company.
- Provides the platform to share media files, important documents using suitable encryption in place.
- Provides the ability to gather user statistics using polls feature.
- Provides platform to create private groups/channel for private messaging within the team.
- Provides the ability for a user to authenticate using 3rd party services such as LinkedIn, Facebook, Google etc.
- Offers different levels of tier services to the client.

Process & Teamwork

We had daily stand-up meetings for about 15-20 minutes to resolve any conflicts and keep a check on the progress. The agenda for the Sprint meeting was as below:

- Identify requirements
 - Decide what needs to be done in order to accomplish a requirement and accordingly take up the following task
- Distribution of work
 - Each team member was assigned issues to work upon keeping in the equal distribution of workload.
- Help teammates
 - If a teammate is stuck on a task, then efforts were made to solve it and help that teammate.

Development Process


- We used Github project for managing our sprints and track our development process

The screenshot displays the GitHub Projects interface for a project named 'FSE Project' (status: Closed, Updated now). The project is located under the repository 'cs5500-fse / team-2-SP20' (Private). The interface includes a top navigation bar with 'Enterprise', a search bar, and links for 'Pull requests', 'Issues', and 'Explore'. Below the navigation bar, there's a 'Project updated' notification. The main area shows a Kanban board with four columns: 'To do', 'In progress', 'Review in progress', and 'Done'. The 'Done' column contains three task cards:

- Media upload - handle on client side** (Issue #39, opened by purvil12c, labeled 'enhancement').
- (Stretch Goal) - Groups - setup, create group, add /remove members, group management** (Added by dipenpatel0810).
- Create MediaService implementation for AWS S3** (Issue #38, opened by purvil12c, labeled 'enhancement').
- Make forward messaging functional from client side** (Issue #36, opened by purvil12c).

Each column has a 'Manage' link at the bottom. The 'To do' column is automated as 'To do', 'In progress' as 'In progress', 'Review in progress' as 'In progress', and 'Done' as 'Done'.

- Wiki page was made documenting all the API

 Enterprise [Pull requests](#) [Issues](#) [Explore](#)

[cs5500-fse / team-2-SP20](#) Private Watch 1 Star 0 Fork 0

[Code](#) [Issues 7](#) [Pull requests 1](#) [Projects 0](#) [Wiki](#) [Insights](#)

C. User related APIs

[Edit](#) [New Page](#)

purvil12c edited this page yesterday · 1 revision

Our client communicates with the Server via HTTP request.

Following is the API documentation for User related endpoints.

SignUp: /user/signup - (POST) body - {username, password, id, name, oAuth}

The user would be able to signup for the Application by sending a POST request to this endpoint. The body of the POST request would contain the username, name, and password for the account. If the user is an OAuth user, then the value of OAuth should be passed as true, if not, then this field can either be false or not included in the body.

Login: /user/login - (POST) body - {username, password}

▼ Pages 12

- [Home](#)
- [A. Introduction](#)
- [B. Building and Deploying Server](#)
- [C. User related APIs](#)
- [D. Message Related APIs](#)
- [E. Messaging through Sockets](#)

- Integrated Github and Jenkins in Slack so that the team would be notified for all the activities.

team-2 ☆

8 | 1 | <https://github.ccs.neu.edu/cs5500-fse/team-2-SP20>



Github APP 12:27 AM

Friday, April 10th

[cs5500-fse/team-2-SP20] Pull request closed: [#185 Poll api fix for deleting response](#) by [j4inam](#)

12:27 [team-2-SP20:master] 3 new commits by purvil12c and 1 other:

[aef13d9](#) Poll api fix for deleting response - purvil12c
[1286662](#) Merge branch 'master' into poll_api_fix - purvil12c
[f0e22fc](#) Merge pull request #185 from cs5500-fse/poll_api_fix - j4inam

[team-2-SP20] The branch "poll_api_fix" was deleted by j4inam



jenkins APP 12:27 AM

SUCCESSFUL: Job 'FSE/team-2-SP20/master'



Github APP 11:40 AM

[cs5500-fse/team-2-SP20] Pull request closed: [#86 Project codebase restructured](#) by [purvil12c](#)

[team-2-SP20] The branch "pulling_out_client" was deleted by purvil12c

[cs5500-fse/team-2-SP20] Issue closed: [#163 user-activity model to be created to store data for tracked users](#) by [purvil12c](#)

[cs5500-fse/team-2-SP20] Issue closed: [#162 APIs for GovernmentRegulations](#) by [purvil12c](#)

[cs5500-fse/team-2-SP20] Issue closed: [#161 toTrack field in UserModel](#) by [purvil12c](#)

[cs5500-fse/team-2-SP20] Issue closed: [#155 Government functions](#) by [purvil12c](#)

[cs5500-fse/team-2-SP20] Issue closed: [#152 User's mute for preset duration for notifications](#) by [purvil12c](#)

[cs5500-fse/team-2-SP20] Issue closed: [#149 in UserController remove v1 URL](#) by [purvil12c](#)

- Implemented Git Workflow


The screenshot displays the IntelliJ IDEA interface with a dark theme. The top toolbar includes buttons for Undo, Redo, Pull, Push, Branch, Stash, Pop, Boards, and Timelines. The left sidebar shows the repository 'team-2-SP20' and the current branch 'master'. Below this, the 'LOCAL' and 'REMOTE' sections are visible, with 'origin' listed as the remote. The 'TAGS' and 'SUBMODULES' sections are also present.



The main panel shows a commit history with a vertical timeline of commits. The commits are color-coded: blue for merges, green for updates, and purple for other changes. The commit messages include:

- Merge pull request #186 from cs5500-fse/purvil12c-patch-1 (12 hours ago)
- Update README.md (yesterday)
- Merge pull request #185 from cs5500-fse/poll_api_fix (a week ago)
- Merge branch 'master' into poll_api_fix
- Merge pull request #183 from cs5500-fse/refactor_2
- Poll api fix for deleting response
- Force Build to run again
- Merge branch 'refactor_2' of https://github.ccs.neu.edu/cs5500-fse/team-2-SP20 into refactor_2
- Force Build to run again
- Merge branch 'master' into refactor_2
- Merge pull request #184 from cs5500-fse/group_controller_modification_2
- Comments added to services package
- comments added to user controller
- Comments added to message controller
- Comments/doc blocks added for govt controller
- error resolved
- changed from 403 to 401
- Private constructor test for ServerSocketConstants added
- missing test added for ClientCommand
- Filled empty exception block
- Removed unused imports from controllers
- Forcing remote SonarQube to run
- Merge branch 'master' into adding_tests
- Merge pull request #181 from cs5500-fse/GovtTests


The right sidebar shows the details of the selected commit (34d758). It includes the commit message 'Merge pull request #186 from cs5500-fse/purvil12c-patch-1', the author 'j4inam' (authored 4/18/2020 @ 9:08 PM), and the committer 'GitHub Enterprise' (committed 4/18/2020 @ 9:08 PM). Below this, it shows '1 modified' file, 'README.md', and options to view the file in 'Path' or 'Tree' view.


- Teammates must review the changes before merging into master

 **purvil12c** wants to merge 1 commit into `master` from `purvil12c-patch-1`



  **purvil12c** requested review from **dipenpatel0810**, **ganatrad** and **j4inam** yesterday


Add more commits by pushing to the **purvil12c-patch-1** branch on **cs5500-fse/team-2-SP20**.





 **Review required** [Hide all reviews](#)


At least 2 approving reviews are required by reviewers with write access. [Learn more](#).

 **3 pending reviewers** 


 **dipenpatel0810** was requested for review

 **ganatrad** was requested for review


 **j4inam** was requested for review

 **All checks have passed** [Show all checks](#)


2 successful checks

 **Merging is blocked**


Merging can be performed automatically with 2 approving reviews.

Merge pull request 


You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Assignees 


No one—assign yourself

Labels 

None yet


Projects 

None yet

Milestone 


No milestone


Notifications [Customize](#)

 **Unsubscribe**

You're receiving notifications because your review was requested.

1 participant



 **Lock conversation**

Job Quality

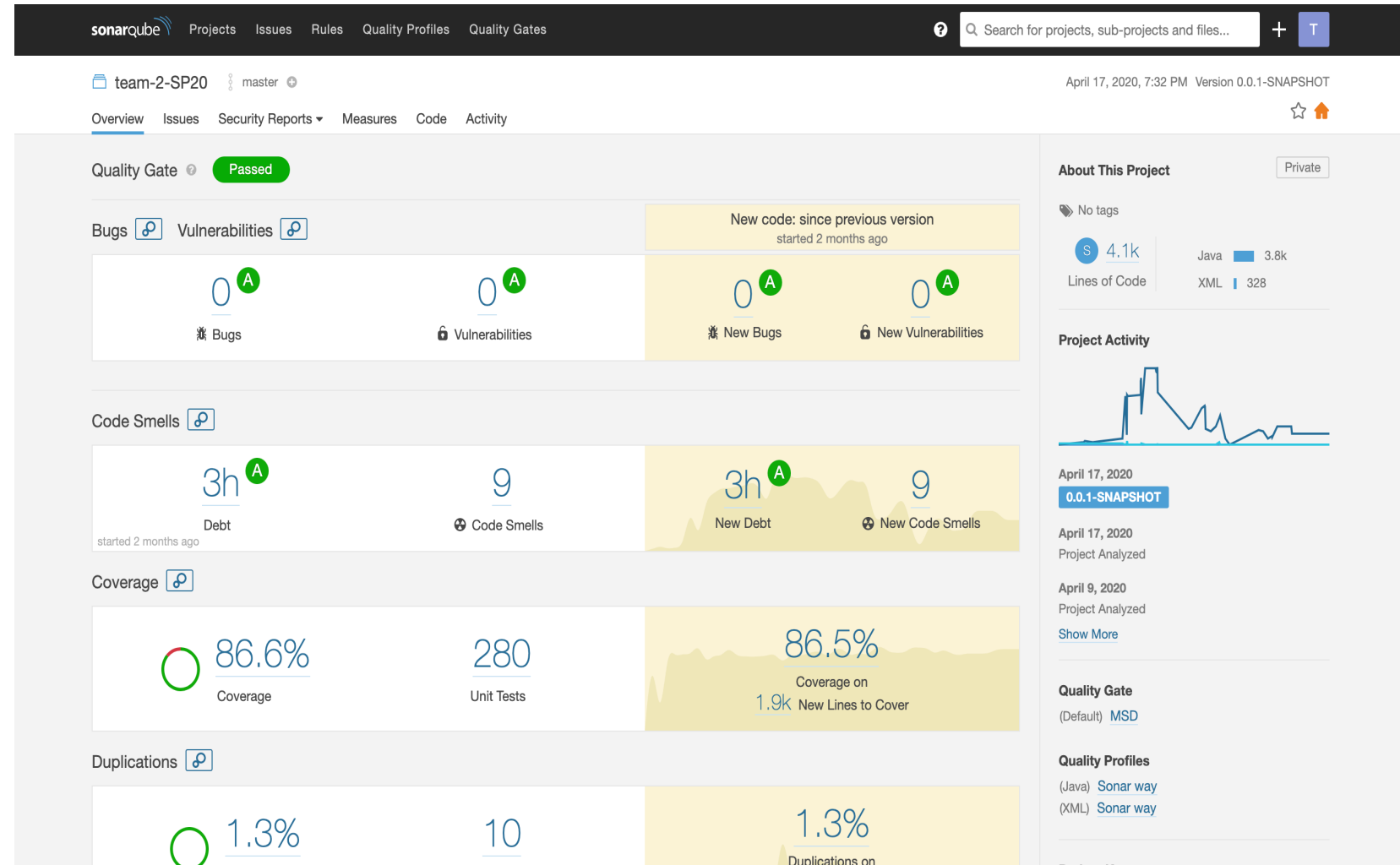
- Database and Services
 - Abstracted in a such a way that a single interface can be extended for any model service.
 - Singleton pattern used to prevent creation of multiple instances.
- Model
 - Builder Pattern used to avoid multiple constructors
- View
 - Observer pattern for client state changes.
 - Command pattern for execution of different commands.

Code Quality

- Code/condition coverage
- Duplications
- Code smells

Testing Mindset

- Feature driven testing
- Unit testing with the help of Mockito
- Integration tests



Next Steps/Improvements

- Areas of Improvement
 - Technical Improvements :
 - Switching to microservice architecture for higher scalability
 - Switching to GraphQL based backend to avoid explosion of controller APIs.
 - Time Management :
 - Avoid last minute tensions across the team

Future Goals

- End to end message translation for better user experience.
- Add time to live for Messages.
- Delete messages for everyone feature.
- Add ability for Parental Control.
- Tracking viewers of the message.

System handover

- System has been intensively tested and has achieved over 85% of code coverage.
- System has been built using recognized design pattern like Builder pattern, Observer pattern etc.
- Best coding practices have been implemented to keep the code modular and reusable. We have followed SonarQube standards to achieve minimal code smells, reduced vulnerabilities etc.
- APIs are extensively documented.

Thank you!