

Sprint 6 Review

Group 6: Ade Aiho, Heta Hartzell , Mika Laakkonen, Jonne Roponen

Statistical Code Review

The group used three statistical code analysis tools, Checkstyle, PMD, and SpotBugs, to analyse the project source code. A comprehensive code review report was written based on the results from these tools. It included an overview and description of the analysed metrics, results and findings from all three tools, and a summary section. Additionally, SonarQube was used to further analyse the code, although its results are not presented in the report.

The purpose of the report was to provide information on what areas of the source code most importantly needed refactoring. The findings showed that multiple classes needed refactoring, and that most issues related to styling problems, bad practices, and bugs.

Code Cleanup

Each member participated in code cleanup through different methods and areas. As an analysis tool was assigned to each member, they could focus on refactoring separate areas and issues, boosting efficiency and preventing duplicate efforts. However, fixing different issues in the same areas of code commonly caused merge conflicts, making the overall process slightly more demanding. Despite this challenge, the code ended up cleaner, more readable, maintainable, and bug free.

Even though most refactoring issues were addressed, it was unclear how far the group had to go during the sprint. Therefore, some refactoring and cleanup was postponed to the next sprint.

Acceptance Testing Planning Report

Acceptance criteria for the acceptance testing was done based on the product backlog user stories. The acceptance testing plan covers functional testing, usability testing, and performance testing. The test cases confirm the core functionality of the software against expected results. Once the acceptance criteria are met in the upcoming sprint, we can say the application is working as expected.

Product Backlog Update

Four new user stories were added to the product backlog, which addressed code clean-up. Users should be able to use the application without any fear of crashes or losing progress, even if they switch views or make some unexpected inputs. Features should also work consistently across the entire application, so the user doesn't get confused in different views and flow stays streamlined. Users also expect the applications to perform all the actions quickly without any lag. Finally the

changes made and refactoring shouldn't result in difference in user experience other than it becoming smoother and more responsive.

The amount of planned story points for this sprint was 14 in total, of which 11/14 were achieved. Updated product backlog is seen in image 1. Only one out of the four user stories was not finished. Due to the acceptance testing plan being only done and not the actual acceptance testing.

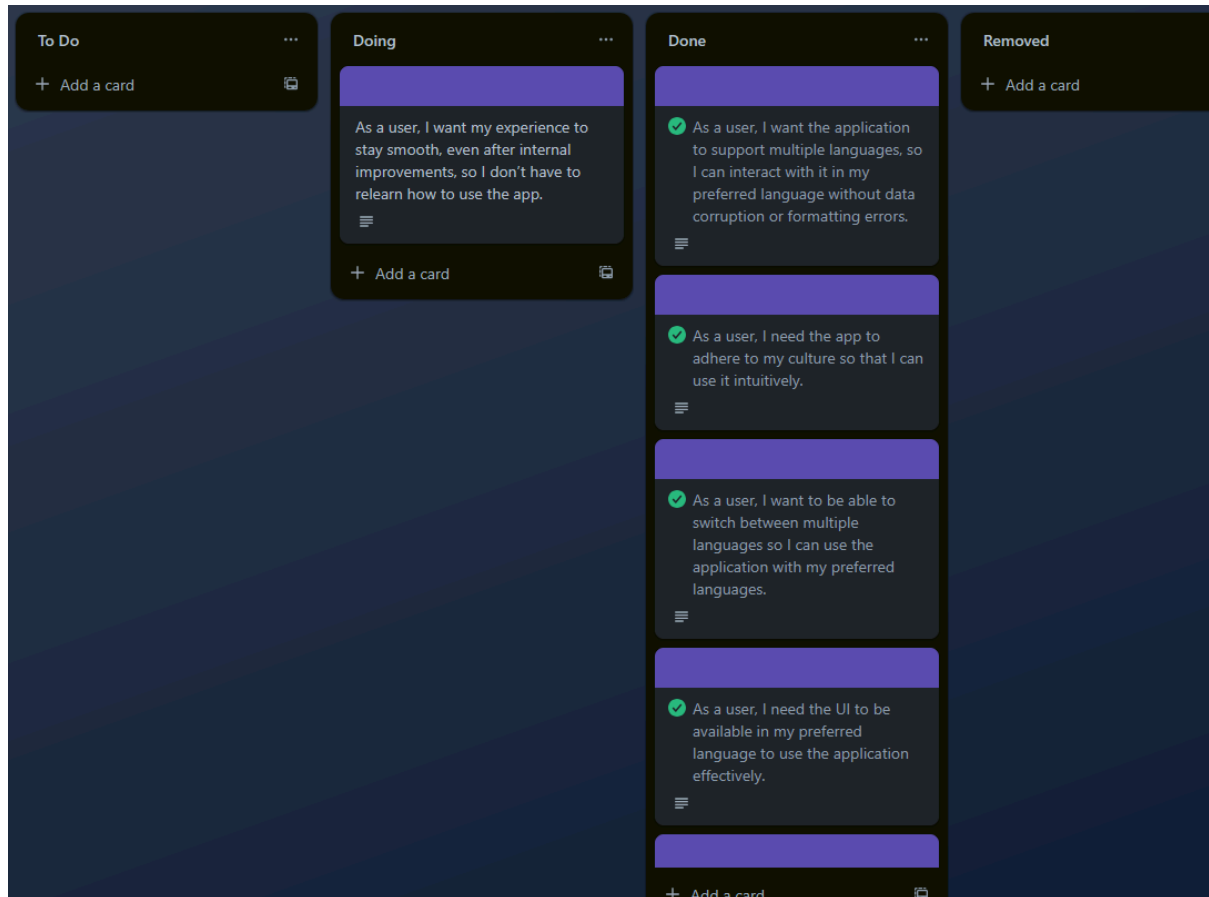


Image 1. sprint 6 product backlog.

The planned user stories in this sprint were split into several tasks that would meet the acceptance criteria of each user story that was marked as done.

Sprint 6 backlog status can be seen in image 2. There were 12 total tasks during this sprint, of which 8 were done and 4 were postponed.

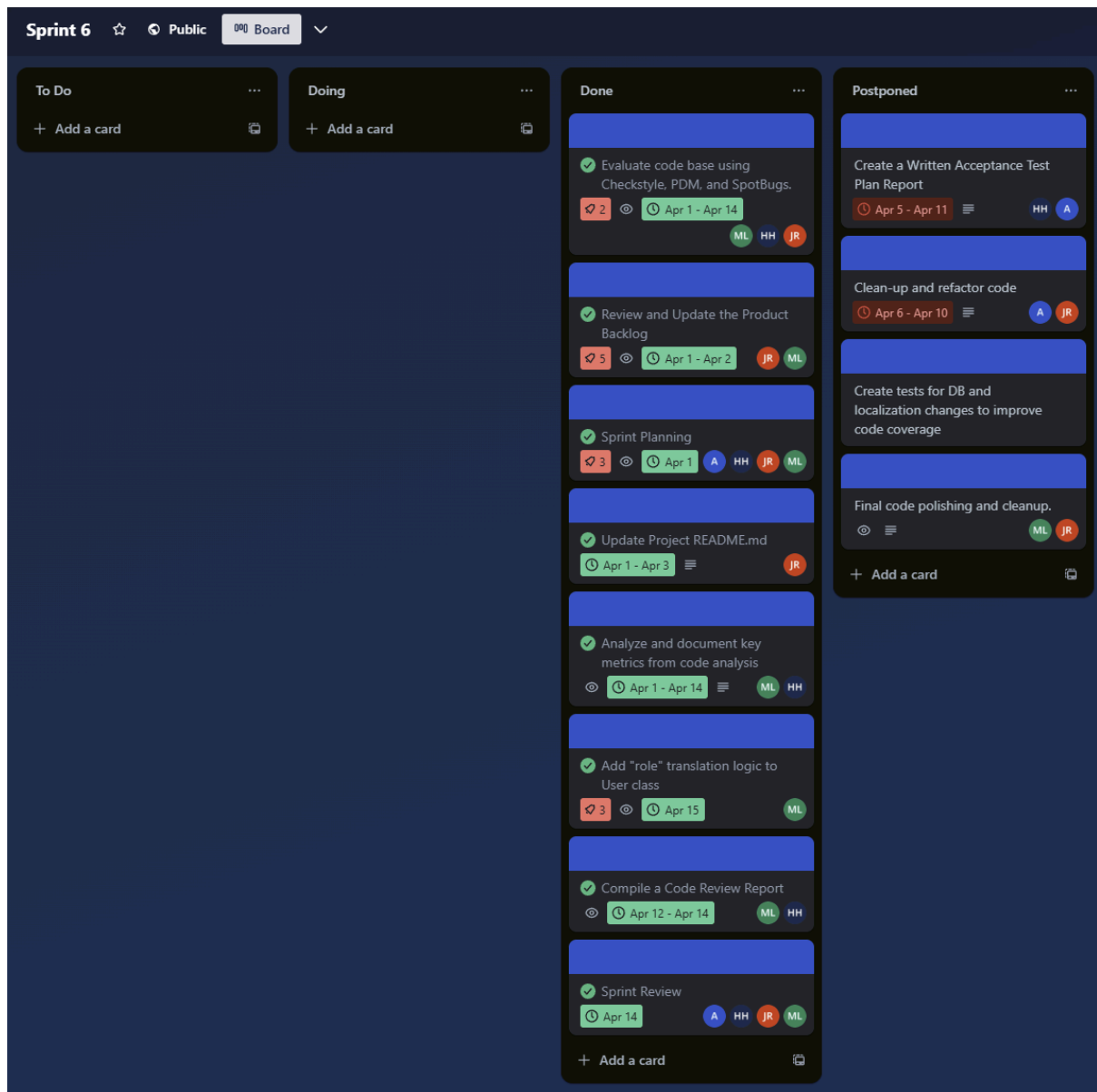


Image 2. sprint 6 backlog.

Sprint Review

The sprint focused on improving code quality, consistency, and performance. Team participated in all sprint ceremonies, the sprint planning and review. After reviewing the sprint 6 requirements the team divided the tasks accordingly between all members.

Project code is now thoroughly analysed with statistical code analysis tools such as checkstyle, PMD, SpotBugs, and SonarQube. Findings were reported in the code review report and improvements and fixes were suggested and implemented. The whole team took part in code refactoring and clean-up to achieve a better function software with smoother user experience.

Name	Tasks	Time
Ade Aiho	Code clean-up, Code refactoring, Bug fixing.	15 hrs
Heta Hartzell	PMD analysis, Code refactoring and cleanup.	15 hrs
Mika Laakkonen	Checkstyle analysis, Sonarqube project setup, Code cleanup and refactoring	18 hrs
Jonne Roponen	SpotBugs analysis for project code, Statistical code review for bugs, Code clean up.	16 hrs

Image 3. shows the excel sheet which displays each team member's contribution.

Ohjelmistotuotantoprojekti 2					
	Sprint 5	Sprint 6	Sprint 7	Sprint 8	Total
Ade	12	15			27
Heta	13	15			28
Mika	12	18			30
Jonne	14	16			30
	51	64	0	0	115

Image 3. image of excel sheet team contributions.