

Summary

During this sprint, we conducted a heuristic evaluation for our project, fixed maintainability issues, fixed unit tests for both frontend and backend and updated some documentation.

Functional Tasks

Functional tasks included the following things:

Testing: Unit tests were fixed for both frontend and backend to fix some remaining issues that came from code cleaning.

For backend that included for example changing some asserts to accommodate the changes made on services.

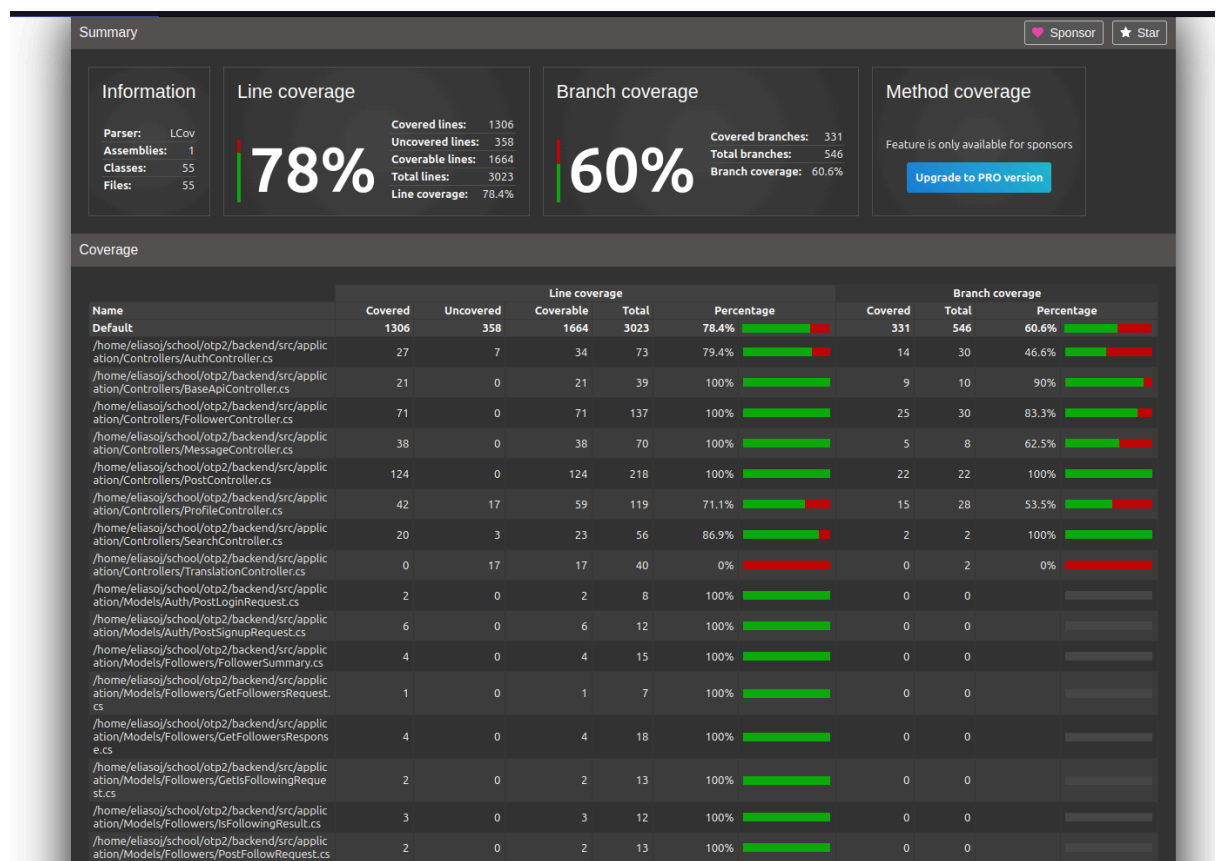
For frontend the changes we also minor and included changing some variables to adhere with the current state of the code the tests were meant for.

Fixing maintainability issues: There were some maintainability issues left that needed fixing for both frontend and backend.

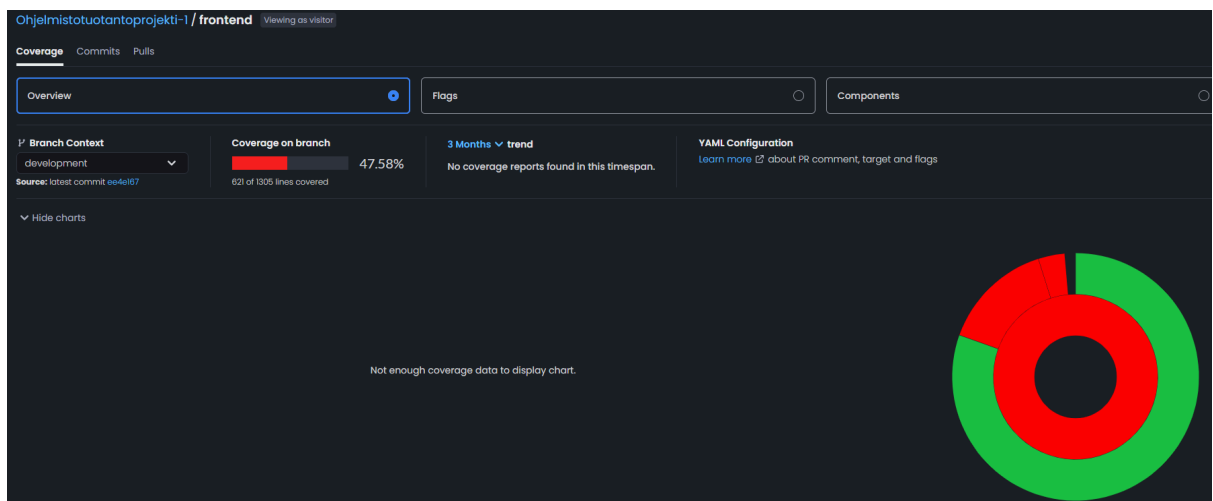
For backend that meant removing some unused variables and refactoring system exceptions to not be thrown by user code. The maintainability issues were reduced from 46 to 13 on sonarqube.

For frontend the changes included removing unused imports, updating component props to Readonly<T> and also refining some internal logic and validation.

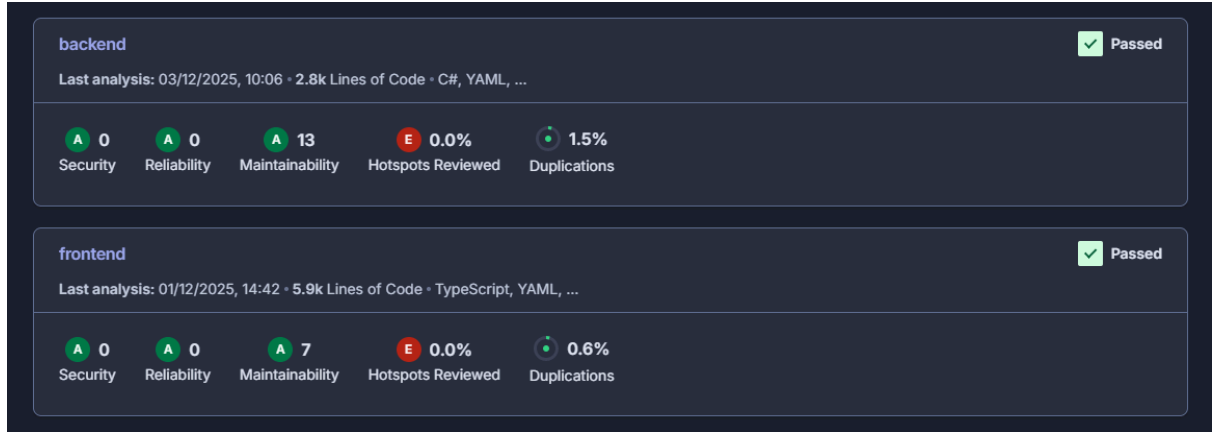
Code coverage for backend:



frontend code coverage:



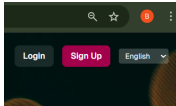
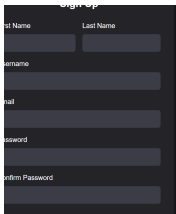
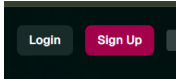
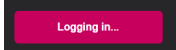
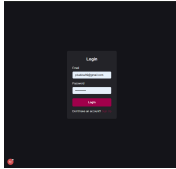
SonarQube:

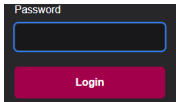
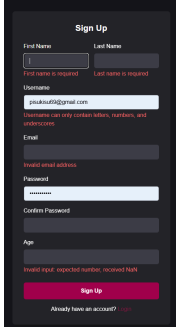
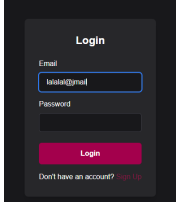


Non-functional Tasks

For non-functional tasks we conducted a heuristic evaluation of our project.

No	Heuristic	Description of the issue	Screenshot	Severity	Suggested improvement
1	H1-1: Simple & natural dialog	The landing page text is vague. "Connect with friends..." doesn't guide the user or explain what to do next.		2	Replace it with a clear call to action like "Sign up to create your profile and start sharing."

2	H1-2: Speak the users' language	The language selector shows “English”, “Suomi”, etc. directly. But if the user doesn’t understand any of those yet, they won’t know it’s a language menu.		2	Replace the label with “Kieli / Language” etc. or use a globe icon. Make it universally understandable before selection.
3	H1-3: Minimize users' memory load	There are no instructions for the user about what kind of data the form needs. The instructions only come after validation (if errors exist)		3	Give instructions about password and username requirements .
4	H1-4: Consistency	“Login” and “Sign Up” buttons are styled differently - one filled, one outlined. It looks unbalanced.		1	Make buttons visually consistent: same size, style, and hover behavior.
5	H1-5: Feedback	The only feedback during login is a static “Logging in...” button. No animation, no spinner, no visual cue that anything is actually happening.		2	Add a loading spinner or subtle animation to show progress. Make it visually clear that the system is working, not frozen.
6	H1-6: Clearly marked exits	No “Cancel” or “Back” option on login view.		1	Add “Back to homepage” or “Cancel” link.

7	H1-7: Shortcuts	There is no “show password” toggle. Typing hidden characters makes it easy to mess up.		2	Add a “show password” option so users can check what they typed before submitting.
8	H1-8: Precise & constructive error messages	Error messages are clear and specific. For an example in the “Sign up” -section.		0	No change needed. Keep this level of clarity across all forms.
9	H1-9: Prevent errors	The site only shows errors after clicking submit. It could warn earlier - like if I type “lalalala@jmail” and jmail isn’t real, it should say “email provider doesn’t exist” or something before I even click.		3	Add live validation while typing. For an example: Check if the email domain is valid and show a warning instantly if it's not.
10	H1-10: Help and documentation	No clear instructions/help button on how to use different parts of the site		2	Add a Help button that would open a modal window that would consist of instructions for the different possibilities the user can do on the site.

Documentations updates

We updated the readme files for both front and backend to include pictures of the current state of sonarcloud (aka sonarQube)

Future plans

In the future we will think about fixing the security hotspots on frontend.