

## Sprint 7 review:

**Group 4: Kapiainen Casper, Phong Le, Vuorela Tino**

### **Done:**

Casper:

- Automated stress
- Functional and reliability tests with playwright
- Code cleanup
- QA documentation

Phong:

- Perform unit testing for all core functionalities
- Perform integration testing for localized features
- Fix bugs identified during testing phase
- Validate SonarQube code quality grades and Capture SonarQube report screenshots
- Perform non functional testing by JMeter

Tino:

- Wrote several frontend unit tests
- Added minor UI/UX improvements
- Code cleanup
- Performed heuristic evaluation

### **In progress:**

- ☹ Casper: Writing playwright tests turned out to take longer than expected; many stress and reliability tests outlined in our acceptance test document still lack implementation.

▼

Sprint 7 18 Nov – 2 Dec (11 work items)

0

0

0

Complete sprint

...

<input checked="" type="checkbox"/>	QTP-99 [Frontend] Performance and reliability testing	DONE ▼	-	CK
<input checked="" type="checkbox"/>	QTP-94 [Frontend] Usability testing	DONE ▼	-	
<input checked="" type="checkbox"/>	OTP-93 [Frontend] Functional testing	IN PROGRESS ▼	-	TV
<input checked="" type="checkbox"/>	QTP-109 [Frontend] UI/UX Improvements	DONE ▼	-	TV
<input checked="" type="checkbox"/>	QTP-102 [Backend] Perform unit testing for all core functionalities	DONE ▼	-	P
<input checked="" type="checkbox"/>	QTP-103 [Backend] Perform integration testing for localized features	DONE ▼	-	P
<input checked="" type="checkbox"/>	QTP-104 [Backend] Fix bugs identified during testing phase	DONE ▼	-	P
<input checked="" type="checkbox"/>	QTP-105 [Backend] Validate SonarQube code quality grades and Capture SonarQube report screenshots	DONE ▼	-	P
<input checked="" type="checkbox"/>	OTP-106 Non-Functional Testing	IN PROGRESS ▼	-	P
<input checked="" type="checkbox"/>	OTP-107 Documentation & QA Process	IN PROGRESS ▼	-	CK
<input checked="" type="checkbox"/>	OTP-108 Sprint 7 Report Deliverables	TO DO ▼	-	

+ Create

Test Result Trend

Build	Passed	Skipped	Failed
#4	0	0	0
#5	0	0	0
#6	0	0	0
#7	0	0	0
#8	10	0	0
#9	30	0	0
#10	35	0	0
#11	100	0	0
#12	100	0	0
#13	100	0	0
#14	100	0	0
#15	100	0	0
#16	100	0	0
#17	100	0	0

Code Coverage Trend

Build	Line Covered	Line Missed
#4	300	400
#5	300	400
#6	300	400
#7	300	400
#8	300	400
#9	450	250
#10	500	200
#11	600	200
#12	600	200
#13	600	200
#14	600	200
#15	600	200
#16	600	200
#17	600	200

[enlarge](#)

Stage View

		Declarative: Checkout SCM	Declarative: Tool Install	Checkout	Build	SonarQube Analysis	Test	Code Coverage	Non-Functional Test	Publish Test Results	Publish Coverage Report
Average stage times: (full run time: ~2min 40s)		1s	872ms	2s	37s	17s	39s	37s	8s	1s	1s
#159	Dec 03 08:01 6 commits	1s	1s	2s	32s	14s	32s	32s	8s	1s	1s
#17	Dec 01 13:45 No Changes	1s	837ms	2s	46s	24s	56s	50s	10s	1s	1s
#16	Dec 01 13:42 1 commit	1s	694ms	2s	36s	18s	31s	28s	8s	1s	1s
#15	Dec 01 13:37 1 commit	1s	502ms	1s	34s	14s	38s	37s	5s failed	580ms failed	530ms failed

main

2k Lines of Code · Version 1.1 · [Set as homepage](#)

✓

Quality Gate ⓘ  
**Passed**

Last analysis 15 seconds ago

⚠ The last analysis has warnings. [See details](#)

New Code

Overall Code

New Code Since November 13, 2025 Started 19 days ago

New issues

0

Required = 0

Accepted issues

0

Valid issues that were not fixed

Coverage

80.0%

Required ≥ 80.0%  
On 222 New Lines to cover.

Duplications

0.0%

Required ≤ 3.0%  
On 2.4k New Lines.

Security Hotspots

0

A

Coverage java in whiteboard (1) (com.otp) ×				
Element ^	Class, %	Method, %	Line, %	Branch, %
com.otp.whiteboard	97% (41/42)	74% (171/230)	78% (513/6...	34% (69/201)
> api	100% (4/4)	100% (20/20)	98% (66/67)	88% (16/18)
> config	100% (5/5)	100% (15/15)	100% (63/63)	100% (0/0)
> dto	100% (13/13)	100% (18/18)	100% (32/32)	100% (0/0)
> enums	100% (4/4)	62% (10/16)	53% (16/30)	0% (0/16)
> listener	100% (1/1)	66% (2/3)	50% (4/8)	100% (0/0)
> model	100% (5/5)	65% (65/100)	54% (84/155)	1% (1/94)
> repository	100% (0/0)	100% (0/0)	100% (0/0)	100% (0/0)
> security	75% (3/4)	20% (4/20)	12% (7/54)	0% (0/16)
> service	100% (5/5)	100% (35/35)	97% (238/2...	91% (52/57)
WhiteboardApplication	100% (1/1)	66% (2/3)	75% (3/4)	100% (0/0)

The report documents the results of static code analysis using PMD to detect type-related violations. The purpose is to improve code safety, maintainability, and adherence to best practices.

Key Metrics:

- Total files analyzed
- Total violations found
- Violation density
- Most frequent violation type

The table below summarizes the type violations detected

The table below summarizes the type violations fixed

## Obstacles:

- None so far.

## Contributions:

Team Member Name	Assigned Tasks	Time Spent (hrs)
Kapiainen Casper	Above	15
Phong Le	Above	15
Tino Vuorela	Above	15

## Links:

Jira:

<https://metropolia-team-4.atlassian.net/jira/software/projects/OTP/boards/1/backlog>

GitHub:

[https://github.com/PhongLe7de/Software\\_Engineering\\_Project\\_1\\_Group\\_4/tree/main](https://github.com/PhongLe7de/Software_Engineering_Project_1_Group_4/tree/main)