

Sprint 3 review:

Group 4: Kapiainen Casper, Phong Le, Vuorela Tino, Tiihonen Leevi

Done:

Leevi:

More work with WebSockets. Establishing real-time functions to work in the app. Drawing / cursor synchronization and history endpoints so that the users can see the drawings in real time.

Casper:

Set up websocket server frontend side. Draw event and client cursor broadcasting and real-time rendering on canvas for all connected clients, cursor components, user auth (register and login) and modal for login/register. Updated app to use context for auth and few other states to minimize prop drilling. Docker image and compose

Phong:

- Redis integration

- Pipeline Configuration

- Jenkin integration

- Board API endpoint integration

- User authentication and API authorization

- Implement Draw event publishing and persistence: publish draw event to Redis channel, store event to Redis list, store draw event to db.

Tino:

Implemented functional drawing tools (pen/eraser, color, brush size), added user info and log out button for the sidebar.

In progress:

The screenshot shows the Jenkins Pipeline Syntax page for a pipeline named 'localhost:9080/job/whiteboard/'. The pipeline is in a 'Stage View' and consists of 8 stages. The stages are: Declarative: Checkout SCM, Declarative: Tool Install, Checkout, Build, Test, Code Coverage, Publish Test Results, and Publish Coverage Report. The Build stage is highlighted in red, indicating a failure. The Test stage is also highlighted in red, indicating a failure. The Code Coverage stage is highlighted in red, indicating a failure. The Publish Test Results stage is highlighted in red, indicating a failure. The Publish Coverage Report stage is highlighted in red, indicating a failure. The Build stage is highlighted in red, indicating a failure. The Test stage is highlighted in red, indicating a failure. The Code Coverage stage is highlighted in red, indicating a failure. The Publish Test Results stage is highlighted in red, indicating a failure. The Publish Coverage Report stage is highlighted in red, indicating a failure.

Stage	Declarative: Checkout SCM	Declarative: Tool Install	Checkout	Build	Test	Code Coverage	Publish Test Results	Publish Coverage Report
Average stage times: (full run time: ~2min 14s)	2s	755ms	2s	40s	33s	19s	879ms	1s
#6 Sep 29 14:59	1s	681ms	2s	31s	38s	46s	1s	2s
#5 Sep 29 14:40	1s	704ms	2s	11s	473ms	473ms	478ms	476ms
#4 Sep 27 16:28	1s	717ms	2s	47s	40s	38s	1s	1s
#3 Sep 27 16:24	1s	803ms	2s	55s	40s	9s	531ms	529ms
#2 Sep 27 16:18	1s	828ms	2s	50s	41s	9s	579ms	531ms
#1 Sep 27 16:10	5s	800ms	2s	49s	41s	10s	550ms	531ms

- None so far

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