

Kartenansicht & Kachel Layout

Code Review

Waldbrandprävention

9. Januar 2023



TECHNISCHE
UNIVERSITÄT
DARMSTADT



User Story

WB-25 Kartenansicht WB-57 Kachel Layout

Entwickler*in

Adrian S., Yannic H.

Reviewer

Yannic H.

Checkliste

- ☐ General
 - ☒ Code is working.
 - ☒ Continuous Integration (CI) is passing.
 - ☒ Code is easily understood.
 - ☒ Code is written following the coding standards/guidelines.
 - ☒ Code is in sync with existing code patterns/technologies.
 - ☒ Same code is not duplicated multiple times. (DRY)
 - ☒ Functions/classes/components are reasonably small.
 - ☒ Naming conventions followed for variables, file names, translations.
 - ☒ Separation of Concerns followed.
- ☐ Codestyle
 - ☐ No hardcoded values, use constants values.
 - ☐ No commented out code.
 - ☒ No unnecessary comments and add necessary comments when needed.
- ☐ Third-Party Libraries
 - ☐ No new npm library was added
 - or -
 - ☒ New npm library was added
 - ☒ Library does not conflict with or break existing code
- ☐ ESLint
 - ☒ Code has no linting errors or warnings.
- ☐ CSS/CSS in JS
 - ☒ No CSS or styled-components used.
 - or -
 - ☐ styled-components is used instead of CSS
- ☐ React/TypeScript
 - ☒ Use const over let where possible and avoid var.
 - ☒ Use function components instead of class components
 - ☒ All React components have proper prop types defined
 - ☒ Components are small.
 - ☒ Variables, functions, and files have clear names and proper cases.
- ☐ Testing

-
- ☐ Neither a Backend story nor API communication involved therefore no manual tests have been created for this story.

- or -

- ☐ Tests are readable and maintainable.
- ☐ Test names are explicit and descriptive.
- ☐ Don't test multiple concerns in the same test.
- ☐ Cover the general case and the edge cases.

○ Git

- ☒ Commits are small and divided into logical parts.
- ☒ Commits messages are small and understandable.
- ☒ Use branches for new features.
- ☒ Make sure no dist files, editor/IDE files, etc are checked in. (.gitignore))

Festgestellte Probleme

- Es sind Hardgecoded Variablen im Code
- Es gibt auskommentierten Code
- Tests fehlen, sind notwendig für die Funktionalität

Konsequenzen

- Hardgecoded Variablen sind kein Problem, da die API noch nicht existiert. Wird später behoben.
- Auskommentierten Code ist für Späteren API Call
- Tests werden, wenn API existiert hinzugefügt

Durchführung

- Test werden im Nachfolge Ticket als Requirement hinzugefügt.
- Hardgecoded Variablen werden im Nachfolge Ticket als Anmerkung hinzugefügt.
- Durchgeführt von Yannic.