# React Refactoring Workshop

February 13th  2026

# What is Refactoring?

The process of restructuring code without changing its external behavior and improve readability.

The goal?

- To transfer knowledge in your head to the code
- To reduce duplication
- To remove unused code
- To extend the life of the project
- To increase speed of new features
- To make onboarding easier
- To increase satisfaction and optimism
- And countless more…

*"Poorly designed code takes more lines to achieve less, obscuring its intent"* – Martin Fowler
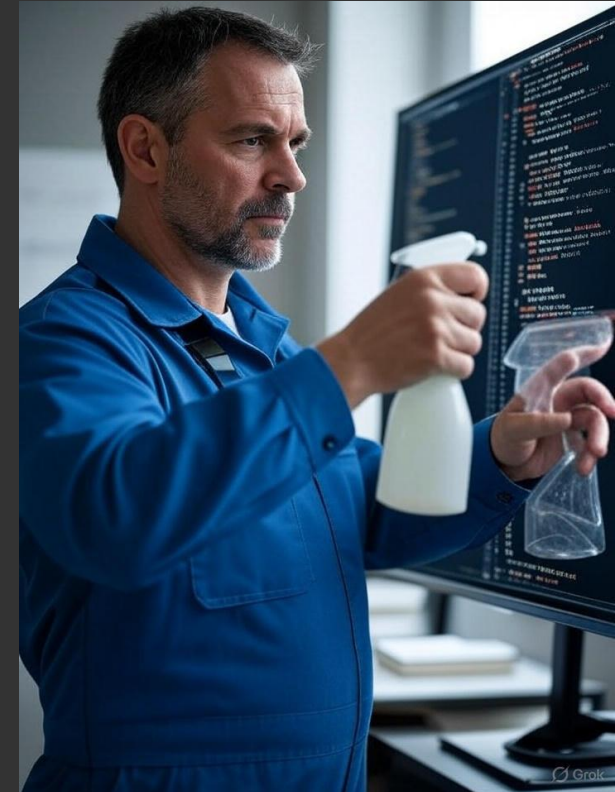
# Clean the code

If you cannot refactor in big swoops, keep an eye out for litter code.

Try to leave components better than you found them

Identifying and eliminating unused code:

- Minimizes potential bugs from outdated logic
- Simplifies codebase for easier navigation
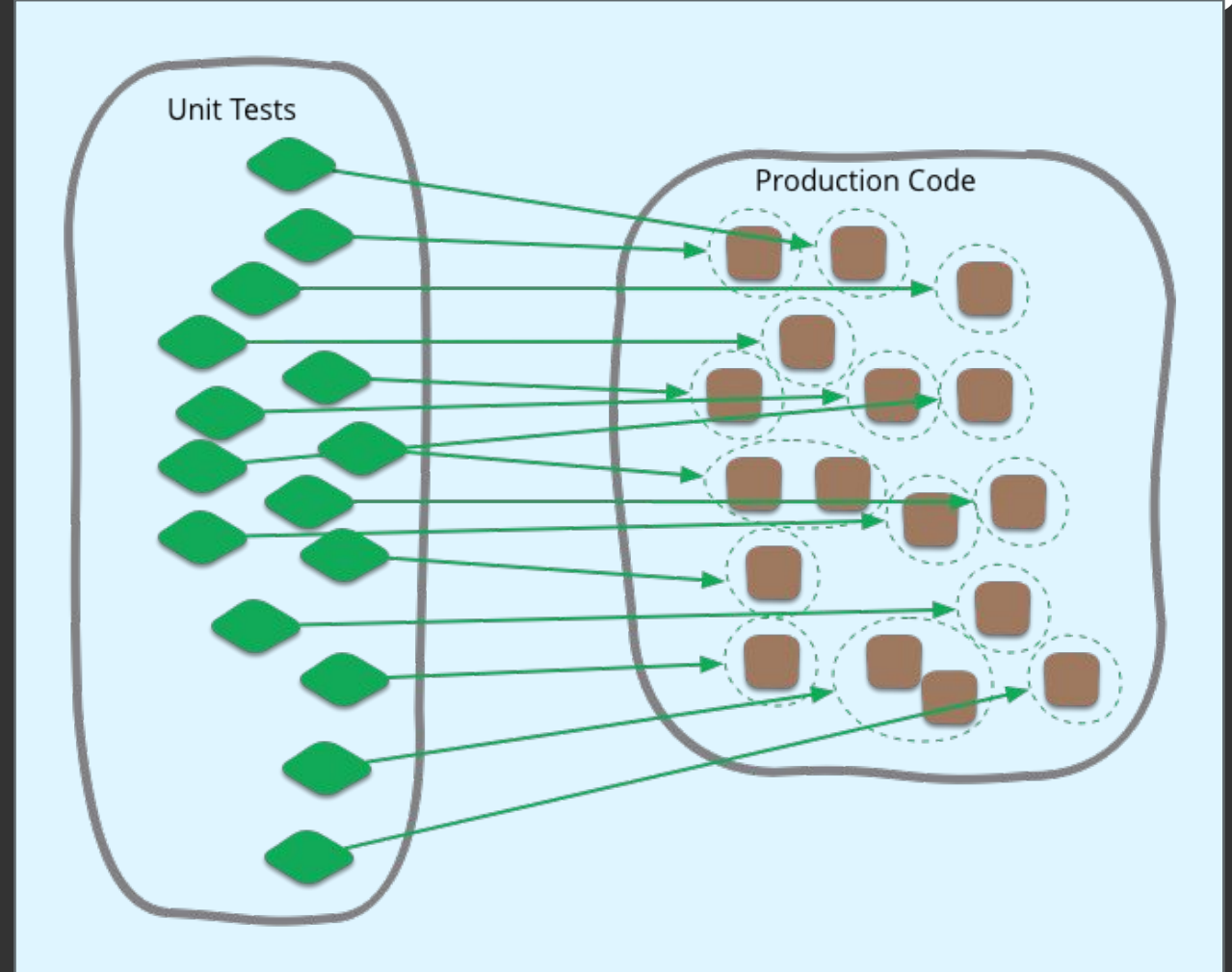- Everything you see is guaranteed to do work

Tools for cleanup in React:

- ESLint + Prettier
- Biome?
- Code analysis tools: **Knip**
- IDE features: "**Find references**" in vscode



*"Litter-pickup refactoring: Clear away the clutter you stumble upon to keep the codebase clean. "* – Martin Fowler

# Unittesting

- Should we write unit tests?
- Enables refactoring to be possible
- Keep them small
- Mock if you have to, speed is important
- A couple of hundred unit tests should take seconds
- Test common utility functions!



https://martinfowler.com/bliki/UnitTest.html

# Unittesting

- Colocation
- Colocation
- Colocation
- Colocation
- Multiple studies show it will have a higher chance of being maintained

## Tools to use?

Vitest + React Testing Library

```
src/
    components/
        Button/
            Button.tsx
            Button.test.tsx
        Header/
            Header.tsx
            Header.test.tsx
```
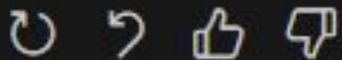
# Copilot and refactoring?

Good when you know React and the codebase

- Bad instructions produce buggy code
- Have an idea of the work to be done
- Then you can write a better prompt
- Provide stacktraces and enough context with the prompt



Thank you for providing me with the stacktrace! I'll analyze the error information to help you resolve the issue.

# The workshop

**Reduce the lines of code!**

- Google Keep clone

- Gamified refactoring

- Gold medal solution located here:

   refactor/notes-app-solution

- Pause the metrics server if your pc starts struggling

- It runs all 107 tests and counting lines continuously

# Where?

https://github.com/DanielTaranger/refactor

- Follow the readme and ask questions

# React Refactoring Workshop

13 Februar 2026

# Hva er refaktorering?

Prosessen med å restrukturere kode uten å endre dens oppførsel og for å øke lesbarhet

Målet?

- Å flytte kunnskapen fra hodet til koden
- Å unngå duplisering
- Å fjerne ubrukt kode
- Å øke levetiden til prosjektet
- Å øke farten til nye saker
- Å gjøre onboarding lettere
- Å skape tilfredshet og optismisme
- Og mye mer…

*"Poorly designed code takes more lines to achieve less, obscuring its intent"* – Martin Fowler
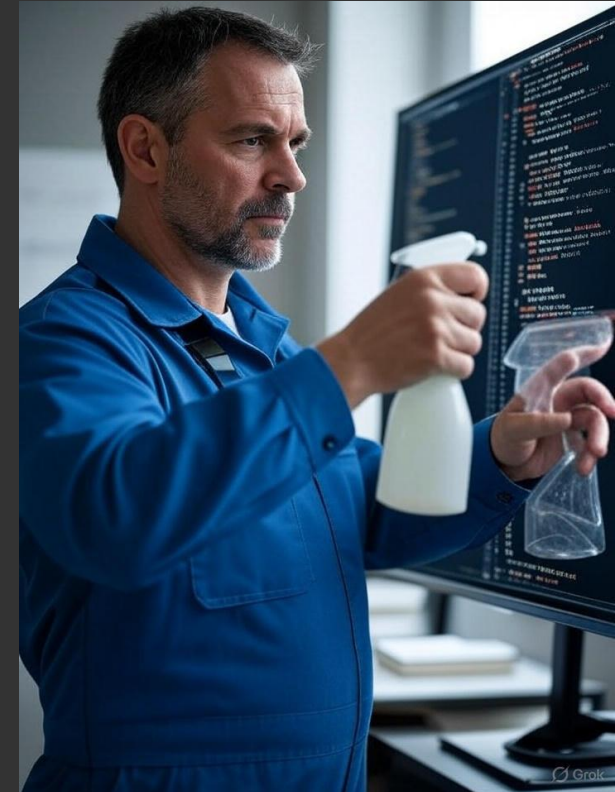
# Rydd opp I koden

Hvis du ikke kan refaktorere i store omganger, pass på å se etter småting.
Prøv å la komponentene være bedre enn du fant dem

Identifiser og eliminer ubrukt kode:

- Minimerer potensielle feil fra utdatert logikk

- Forenkle kodebasen for enklere navigering

- Alt du ser er garantert å gjøre arbeid

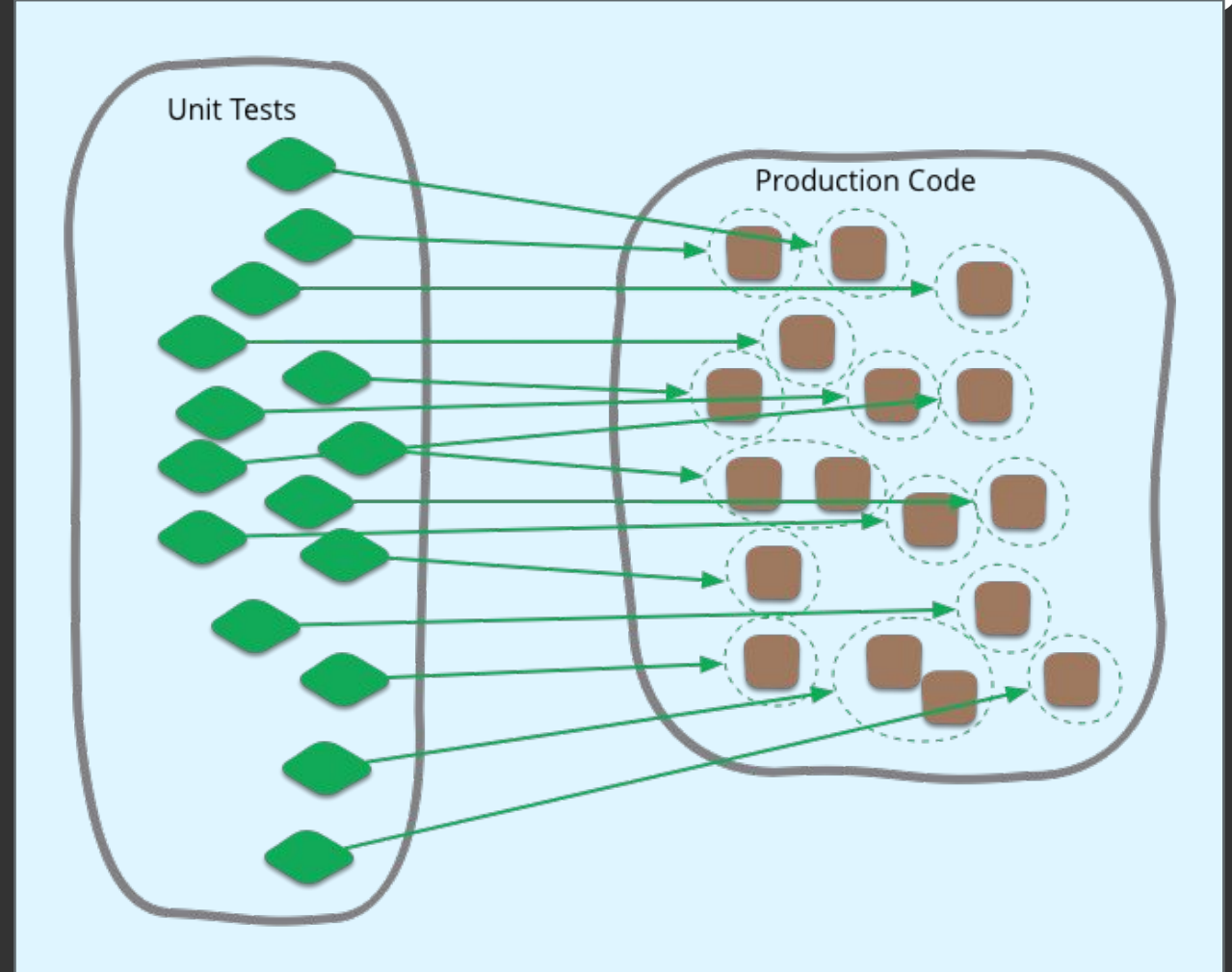Verktøy for opprydding i React:

- ESLint + Prettier

- Biome?

- Kode analyseverktøy **Knip**

- IDE features: "**Find references**" in vscode

*"Litter-pickup refactoring: Clear away the clutter you stumble upon to keep the codebase clean. "* – Martin Fowler

# Unittesting

- Bør vi skrive unittester?
- Gjør refaktorering mye lettere
- Hold testene små
- Mock hvis du må
- Et par hundre unittester burde ta sekunder
- Test common utility funksjoner

Unit Tests

Production Code

https://martinfowler.com/bliki/UnitTest.html

# Unittesting

- Kollokasjon
- Kollokasjon
- Kollokasjon
- Kollokasjon
- Flere studier viser at kode vil ha høyere sjanse for å bli vedlikeholdt

## Verktøy

Vitest + React Testing Library

```
src/
    components/
        Button/
            Button.tsx
            Button.test.tsx
        Header/
            Header.tsx
            Header.test.tsx
```
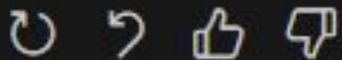
# Copilot og refaktorering

Bra når du kjenner React og kodebasen

- Dårlige instruksjoner gir (ofte) dårlig kode
- Ha en ide om hva copilot skal gjøre
- Så kan du lettere skrive en god prompt
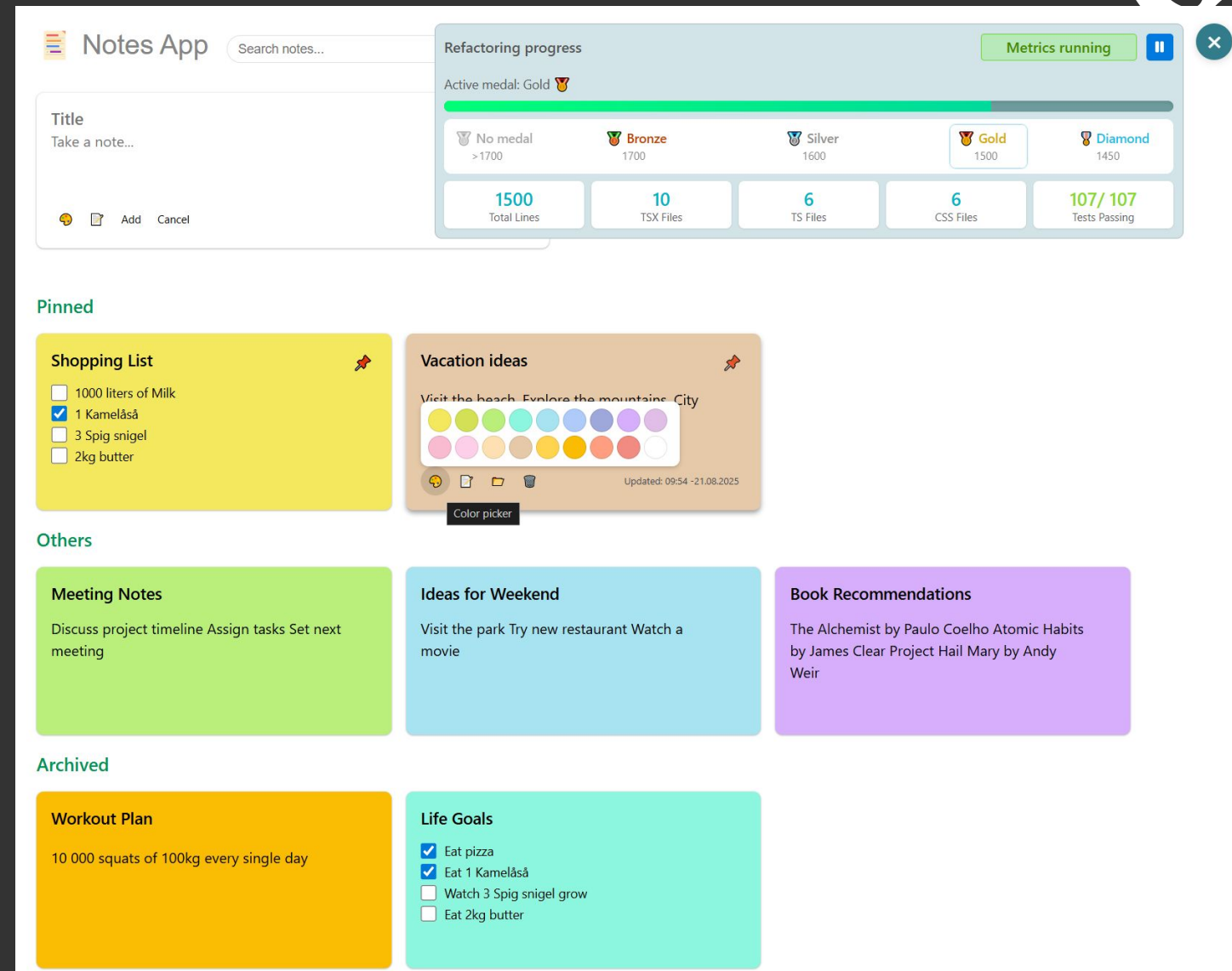- Gi den feilmeldinger og nok kontekst sammen med prompten



Thank you for providing me with the stacktrace! I'll analyze the error information to help you resolve the issue.

# Workshop

**Reduser antall linjer med kode!**

• Google Keep klone

• Gamified refaktorering

• Gull medalje løsning ligger I denne mappen:

   refactor/notes-app-solution


• Pause metrics serveren hvis pcen din sliter

• Den kjører 107 tester kontinuerlig og teller kode

# Where?

https://github.com/DanielTaranger/refactor

- Follow the readme and ask questions