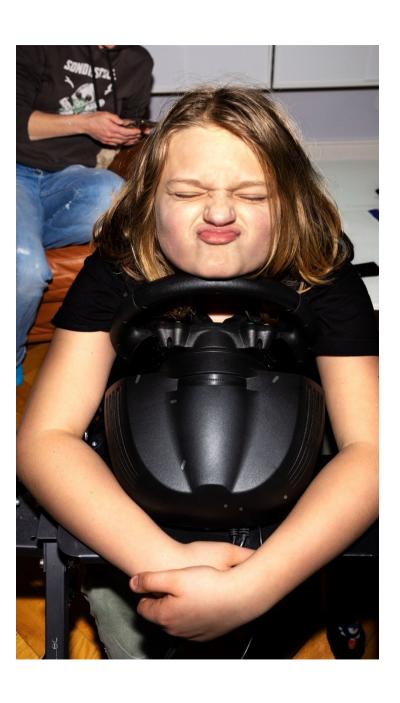


# Have you gone MADR? Johan Hutting, Engineer / developer advocate @ ING





# Why did I/we decide this again?

#### Over time it gets harder to remember

- Agile manifesto:
   "Working software over comprehensive documentation"
- How well do your handovers go?
- A lot of information is in our heads, fading over time
- Or in Email chains can you still find it?
- And in chats, where searching is even harder
  - Or history is deleted after x years

# MarkDown Any Decision as a Record How does that help us?

# **Documenting decisions**

#### Source - https://adr.github.io/madr/

- Designed for architectural decisions, along with code
- At a minimum document the
  - Decision made
  - Context and problem statement
  - Considered options
  - Outcome detailing pro's/cons





## **Applying MADR**

#### The Agile way: document when it adds value

- DRY link to extensive resources.
- Document alternatives and reasoning
- Decisions should be mostly static
  - Different insights over time? New MADR!
- Add decisions when you think or someone mentions:
   "this might be useful for future reference"
- Review them like code pull requests
- The above should smoothen future discussions
- Don't add all historical decisions at once, take your time



# Java 23+ JavaDoc with Markdown

#### JEP 467: Markdown Documentation Comments

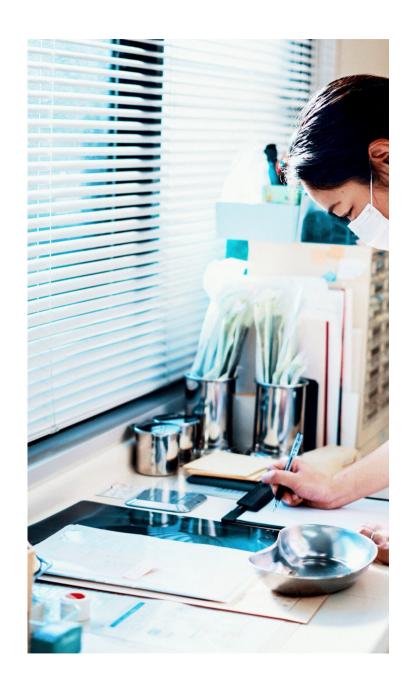
Make API documentation comments easier to write and easier to read in source form by introducing the ability to use Markdown syntax in documentation comments, alongside HTML elements and JavaDoc tags.

Key benefit: use the same markup language for both decisions and elaborating code.

### Recap

#### Focus on important decisions in an accessible format

- Document the important decisions in a concise way
  - Link to larger, more in-depth resources
- Have them in one markup along with your code
- Align with your fellow developers on what works well





# Thank you!