Untangled:

from Spaghetti Code to Masterpiece

In 6 easy steps!
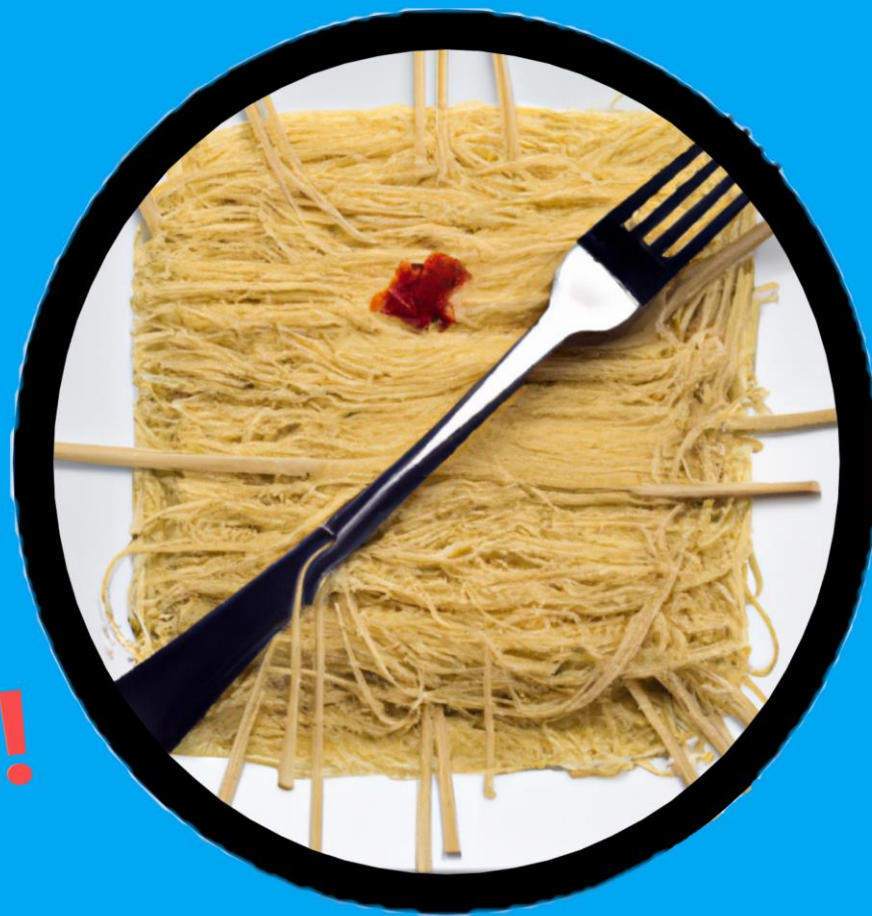
Refactoring Legacy Code

BUT WHY?

Why?

- Easy understanding

- Maintainability

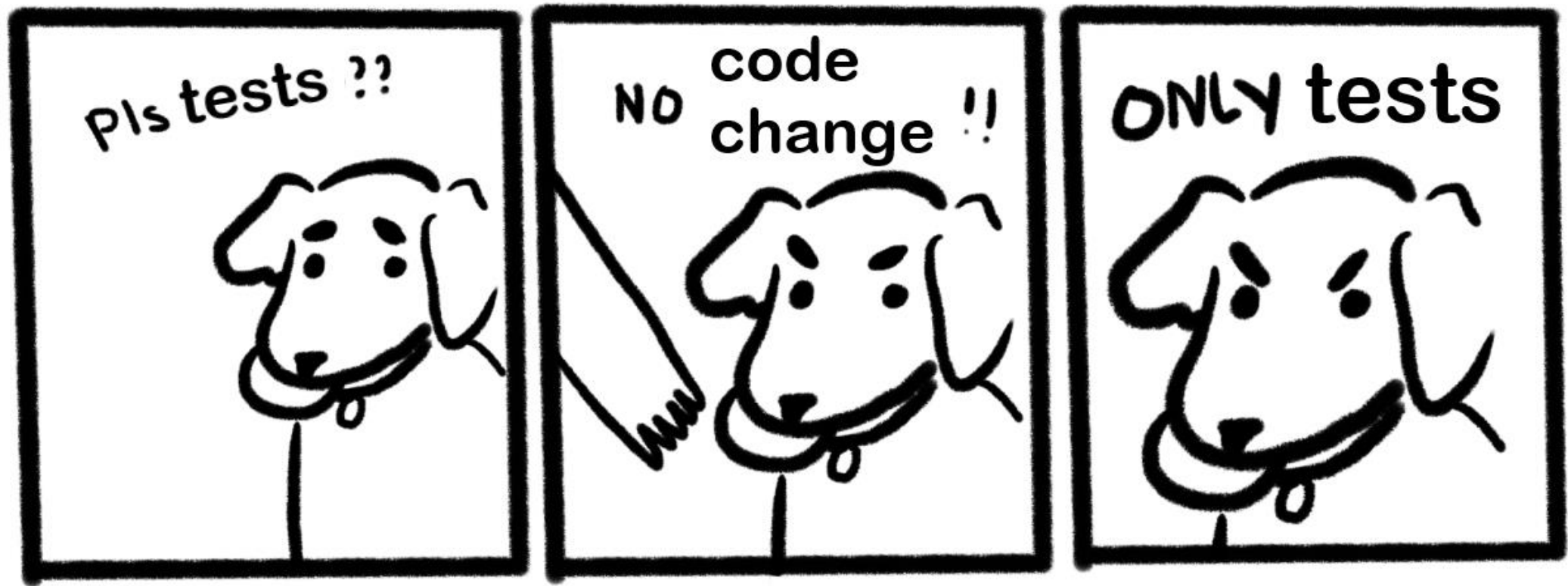- Reduce the risk of bugs

- May improve performance

HOW?

Step 1: Understanding the codebase

- Reviewing

- Identifying dependencies

- Interaction between components

- Analyzing behavior

Step 2: Find areas for improvement

- Simplifying complex code

- Removing redundancy

- Restructure code

Step 3: Tests, tests, tests

Step 4: Refactoring

- Start refactoring

- Ensure tests are passing

- Behavior remains the same

Step 5: Test and validate

- Execute test suite

- Validate correct functionality

- Automatic and/or manual tests, depending on complexity

Step 6: Repeat

- .
- .
- .
- .

refactoring

The definition of ~~insanity~~ is doing the same thing over and over again and expecting ~~a different~~ result.

the same test-

- probably not Albert E.

Conclusion

- Refactoring requires thought and understanding

- A comprehensive test suite

- Effort

Thanks for your attention!

Feel free to reach me at:

   markus.reiss@betterask.erni

•

Sources:
- Title: Dall-E generated
- Why: https://imgflip.com/memegenerator/58542746/But-Why
- How: https://imgflip.com/memegenerator/44529090/Jackie-Chan-Confused
- Step 3: https://cupcakelogic.tumblr.com/post/124392369931/she-is-still-learning
- Step 6: https://quoteinvestigator.com/2017/03/23/same/
- Final Page: https://imgflip.com/memegenerator/Doge