

The background features a dark blue gradient with a subtle pattern of white stars. Overlaid on this are several technical diagrams in a lighter blue color. These include circular gauges with numerical scales (e.g., 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260), concentric circles, and curved arrows indicating motion or flow. The diagrams are semi-transparent and scattered across the left and top portions of the frame.

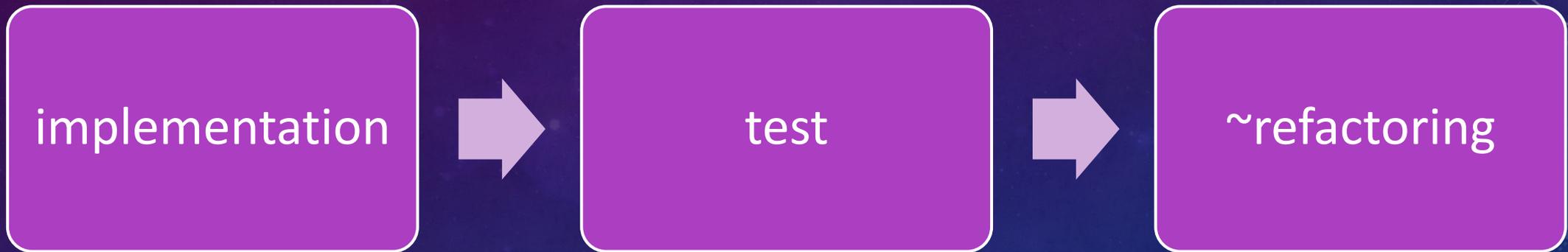
TEST CODE IMPROVEMENTS

DÁNIEL POLGÁR

FACTS ABOUT TESTING

- Ensure functionality
- Massive reduction of bugs
- 62,3% of developers write tests (2019)
- Unpopular

WORKFLOW



5 Methods (2 public)

multiple test methods

can be very difficult

PROBLEMS WITH TESTING

- Need extra work & time
- Hard to find names
- Not always usable or difficult
- Find the correct depth
- Focus on test coverage

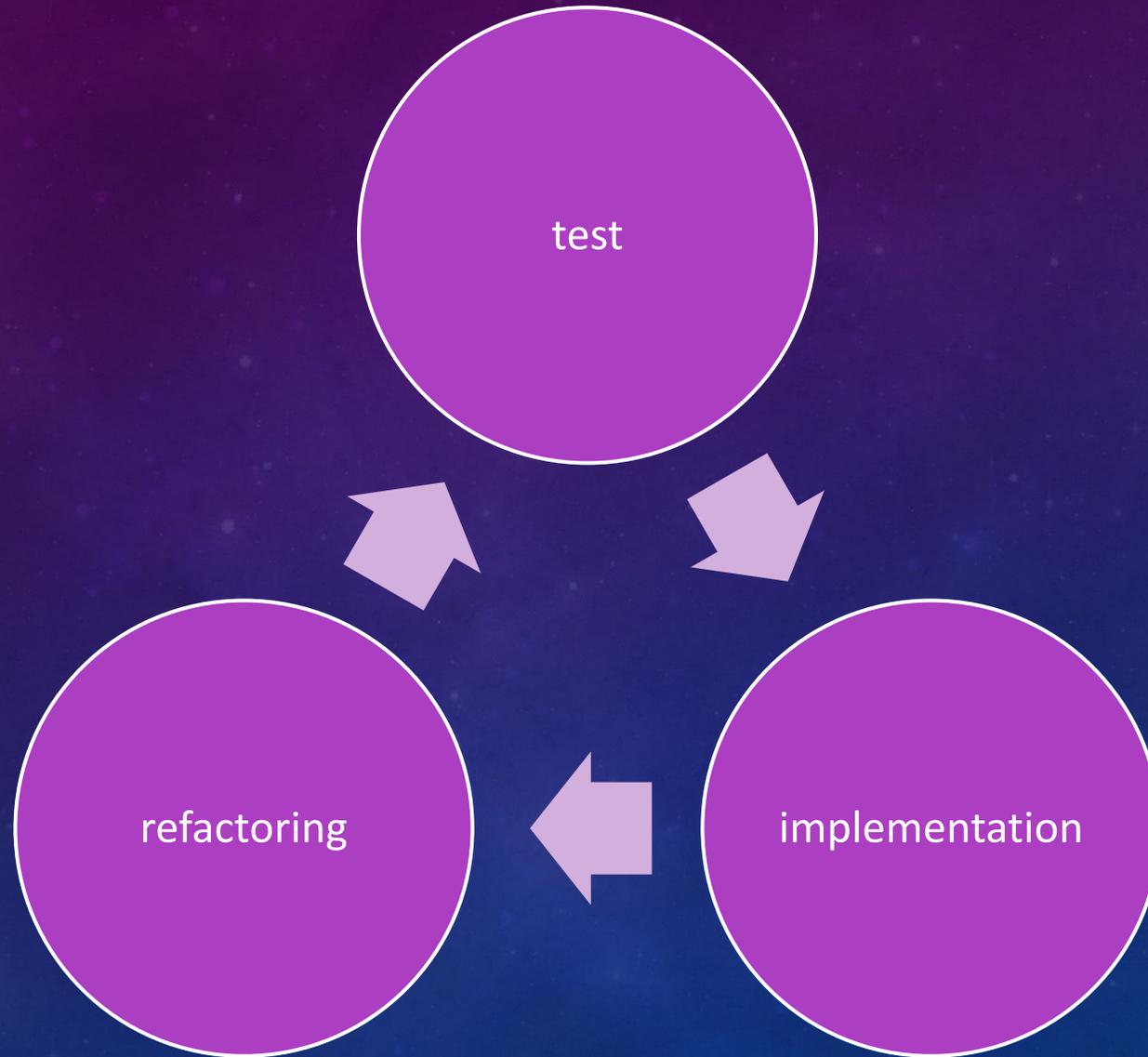
=> less motivation

IMPROVEMENTS

- TDD
- IDE
- Micro commits
- Mob or pair

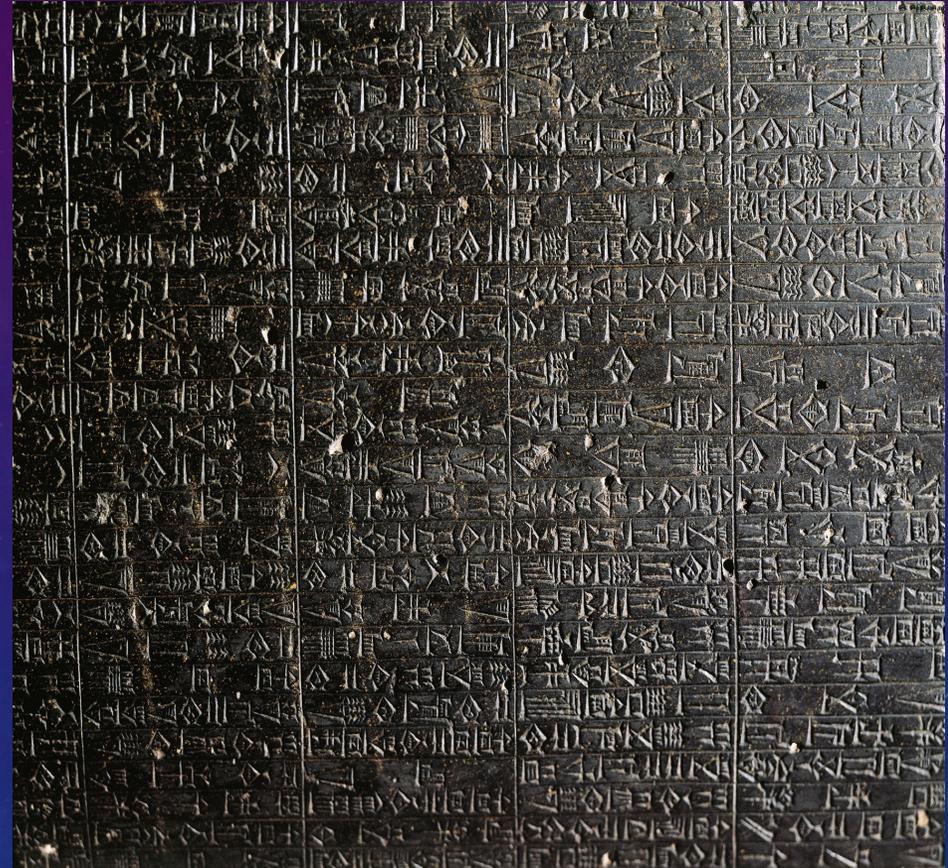


TDD – CYCLE



TDD - RULES

- Run the test always after created
-  feature &  implementation
- Always the least code to make the test green
- Assert -> actual -> arrange



TDD - BENEFITS

- Easier to write clean tests and code
- Guarantees testable code
- Avoids too large steps
- The workflow of the developer is consistent
- Documents the code

IDE

- Shortcuts/hints
 - Variables
 - Methods
 - Classes
 - Executions
- Warnings
- Code formatting
- Git



MICRO COMMITS

- Commit after each step
- Mark the commits with an capital letter
 - F -> feature
 - T -> tests
 - R -> refactoring
- Clean commit history

MOB & PAIR

- More ideas
- Needs no review
- Sharing the problem



CONCLUSION OF THE IMPROVEMENTS

- Need extra work & time ✓
- Hard to find names ✓
- Not always usable or difficult ✓
- Find the correct depth ✓
- Focus on test-coverage ✓



THANK YOU!

