

# Software Risk Management: Code Review

17-313 Fall 2023

Foundations of Software Engineering

<https://cmu-313.github.io>

Andrew Begel and Rohan Padhye

# Administrivia

- Mid-term exam next week (Oct 10) in class
- Recitation this week: midterm review (**come prepared!**)
  - <https://cmu-313.github.io/recitations/reci6-midterm-review/>
  - Work through problems on the previous midterms – many students found this helpful.
  - Any questions on the previous midterm questions – bring them to recitation to discuss as a class.
- Fill in Team Assessment Survey by Friday 3:00pm
- Final Presentations (P5):  
Tuesday December 12<sup>th</sup>, 5:30 pm - 8:30pm, Room TBD

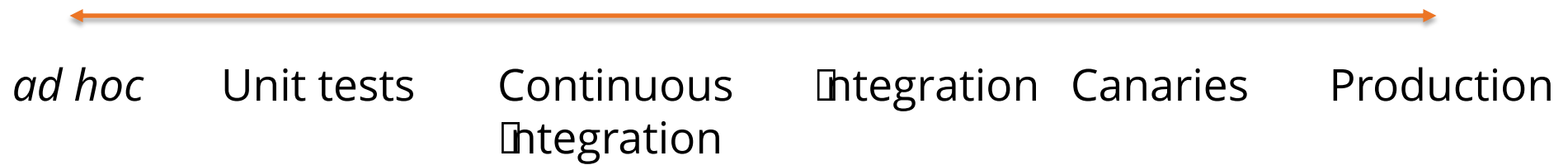
# Administrivia

- Participation exercises are meant for in-class participation only. Please do not submit anything to Gradescope if you are not physically present in PH 100.

# Ways to Test and Validate Your Code

- Static Validation
  - Stare at the code
- Dynamic Validation
  - Run the source code

# Dynamic Validation



# Static Validation

- Style guides
- Compiler warnings and errors
- Static analysis
  - [FindBugs](#)
  - [clang-tidy](#)
  - [Pylons Webtest](#)
- Code review

# Style Guide

- List of environment-specific preferred practices
- Could include:
  - Libraries / idioms to use
  - Formatting

# Style Guide Examples

- <https://www.python.org/dev/peps/pep-0008/>
- <https://github.com/airbnb/javascript>
- <https://subversion.apache.org/docs/community-guide/conventions.html>
- <https://google.github.io/styleguide/cppguide.html>
- <https://google.github.io/styleguide/pyguide.html>
- [Linux kernel style guide](#)



# Who writes these style guides?



# Who writes these style guides?

*(ad hoc 🍑🧠)* Self-proclaimed code protectors

*(wisdom)* Team veteran developers

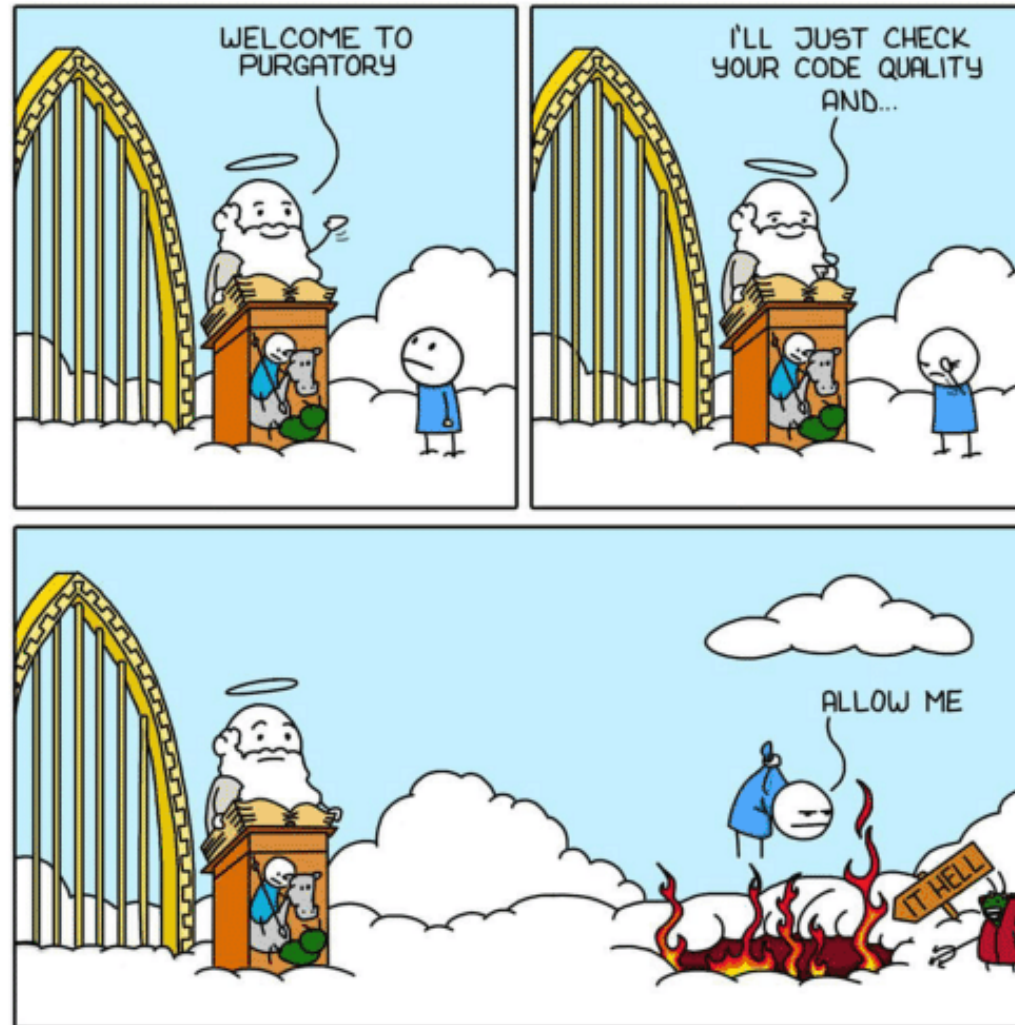
*(copy-paste)* Google search for blog posts by experts

*(empirical study)* Evidence-based analysis of code styles that correlate with bugs

# Code Review

- Does this code do what it claims?
  - Are there any programming bugs?
- Why are we making this change?
  - Are there any design bugs?

# LAST PUSH



MONKEYUSER.COM

last push

# Fishy Code Example #1

```
7  class Test
8  {
9
10     public function doSomething($param)
11     {
12         // Do something
13     }
14
15     public function doSomethingOther(int $param): void
16     {
17         // Do something other
18     }
19
20     protected function internalMethod($param)
21     {
22         // Do something else
23     }
24 }
```

# Fishy Code Example #2a

```
7  class Test
8  {
9
10     public function doSomeFormatting($input)
11     {
12         if (empty($input)) {
13             return;
14         }
15
16         $result = strtoupper(trim($input));
17         return $result;
18     }
19 }
```

# Fishy Code Example #2b

```
public function doSomeFormatting(string $input): string
```

# Fishy Code Example #2c

```
7  class Test
8  {
9
10     public function doSomeFormatting(string $input): string
11     {
12         return strtoupper(trim($input));
13     }
14
15 }
```



# Checklists help manage complex processes



**OFFICIAL A.A.F. PILOT'S CHECK LIST**  
B-25 AND B-25C  
For detailed instructions see Pilot's Handbook 401-21-2001 or  
401-21-2001a in same issue.

| PILOT  | CO-PILOT  |
|--|---|
| <b>BEFORE STARTING</b>   | <b>BEFORE TAKE OFF</b>                              |
| 1. Main Fuel Valve - <input type="checkbox"/> Open   | 1. Tail Wheel - <input type="checkbox"/> Locked     |
| 2. Power (A, B, C) - <input type="checkbox"/> High and <input type="checkbox"/> Balance - <input type="checkbox"/> Checked | 2. Brakes - <input type="checkbox"/> OK             |
| 3. Controls and Instruments - <input type="checkbox"/> Checked   | 3. Instruments - <input type="checkbox"/> OK        |
| 4. Fuel Transfer Valves and Switches - <input type="checkbox"/> OK   | <b>AFTER TAKE OFF</b>                               |
| 5. Intercom - <input type="checkbox"/> OK  | 1. Wheels - <input type="checkbox"/> Fully Expanded |
| 6. Air - <input type="checkbox"/> On   | 2. Power - <input type="checkbox"/> OK              |
| 7. Fuel Boost - <input type="checkbox"/> On  | 3. Fuel - <input type="checkbox"/> OK               |
| 8. Fuel Boost - <input type="checkbox"/> On  | 4. Wheel Chock - <input type="checkbox"/> Off       |
| 9. Fuel Boost - <input type="checkbox"/> On  | <b>BEFORE LANDING</b>                               |
| 10. Fuel Boost - <input type="checkbox"/> On   | 1. Radio Call - <input type="checkbox"/> OK         |
| 11. Fuel Boost - <input type="checkbox"/> On   | 2. Fuel - <input type="checkbox"/> OK               |
| 12. Fuel Boost - <input type="checkbox"/> On   | 3. Air - <input type="checkbox"/> OK                |
| 13. Fuel Boost - <input type="checkbox"/> On   | 4. Brakes - <input type="checkbox"/> OK             |
| 14. Fuel Boost - <input type="checkbox"/> On   | 5. Hydraulic - <input type="checkbox"/> OK          |
| 15. Fuel Boost - <input type="checkbox"/> On   | 6. Landing Gear - <input type="checkbox"/> OK       |
| 16. Fuel Boost - <input type="checkbox"/> On   | 7. Fuel - <input type="checkbox"/> OK               |
| 17. Fuel Boost - <input type="checkbox"/> On   | 8. Fuel - <input type="checkbox"/> OK               |
| 18. Fuel Boost - <input type="checkbox"/> On   | 9. Fuel - <input type="checkbox"/> OK               |
| 19. Fuel Boost - <input type="checkbox"/> On   | 10. Fuel - <input type="checkbox"/> OK              |
| 20. Fuel Boost - <input type="checkbox"/> On   | 11. Fuel - <input type="checkbox"/> OK              |
| 21. Fuel Boost - <input type="checkbox"/> On   | 12. Fuel - <input type="checkbox"/> OK              |
| 22. Fuel Boost - <input type="checkbox"/> On   | 13. Fuel - <input type="checkbox"/> OK              |
| 23. Fuel Boost - <input type="checkbox"/> On   | 14. Fuel - <input type="checkbox"/> OK              |
| 24. Fuel Boost - <input type="checkbox"/> On   | 15. Fuel - <input type="checkbox"/> OK              |
| 25. Fuel Boost - <input type="checkbox"/> On   | 16. Fuel - <input type="checkbox"/> OK              |
| 26. Fuel Boost - <input type="checkbox"/> On   | 17. Fuel - <input type="checkbox"/> OK              |
| 27. Fuel Boost - <input type="checkbox"/> On   | 18. Fuel - <input type="checkbox"/> OK              |
| 28. Fuel Boost - <input type="checkbox"/> On   | 19. Fuel - <input type="checkbox"/> OK              |
| 29. Fuel Boost - <input type="checkbox"/> On   | 20. Fuel - <input type="checkbox"/> OK              |
| 30. Fuel Boost - <input type="checkbox"/> On   | 21. Fuel - <input type="checkbox"/> OK              |
| 31. Fuel Boost - <input type="checkbox"/> On   | 22. Fuel - <input type="checkbox"/> OK              |
| 32. Fuel Boost - <input type="checkbox"/> On   | 23. Fuel - <input type="checkbox"/> OK              |
| 33. Fuel Boost - <input type="checkbox"/> On   | 24. Fuel - <input type="checkbox"/> OK              |
| 34. Fuel Boost - <input type="checkbox"/> On   | 25. Fuel - <input type="checkbox"/> OK              |
| 35. Fuel Boost - <input type="checkbox"/> On   | 26. Fuel - <input type="checkbox"/> OK              |
| 36. Fuel Boost - <input type="checkbox"/> On   | 27. Fuel - <input type="checkbox"/> OK              |
| 37. Fuel Boost - <input type="checkbox"/> On   | 28. Fuel - <input type="checkbox"/> OK              |
| 38. Fuel Boost - <input type="checkbox"/> On   | 29. Fuel - <input type="checkbox"/> OK              |
| 39. Fuel Boost - <input type="checkbox"/> On   | 30. Fuel - <input type="checkbox"/> OK              |
| 40. Fuel Boost - <input type="checkbox"/> On   | 31. Fuel - <input type="checkbox"/> OK              |
| 41. Fuel Boost - <input type="checkbox"/> On   | 32. Fuel - <input type="checkbox"/> OK              |
| 42. Fuel Boost - <input type="checkbox"/> On   | 33. Fuel - <input type="checkbox"/> OK              |
| 43. Fuel Boost - <input type="checkbox"/> On   | 34. Fuel - <input type="checkbox"/> OK              |
| 44. Fuel Boost - <input type="checkbox"/> On   | 35. Fuel - <input type="checkbox"/> OK              |
| 45. Fuel Boost - <input type="checkbox"/> On   | 36. Fuel - <input type="checkbox"/> OK              |
| 46. Fuel Boost - <input type="checkbox"/> On   | 37. Fuel - <input type="checkbox"/> OK              |
| 47. Fuel Boost - <input type="checkbox"/> On   | 38. Fuel - <input type="checkbox"/> OK              |
| 48. Fuel Boost - <input type="checkbox"/> On   | 39. Fuel - <input type="checkbox"/> OK              |
| 49. Fuel Boost - <input type="checkbox"/> On   | 40. Fuel - <input type="checkbox"/> OK              |
| 50. Fuel Boost - <input type="checkbox"/> On   | 41. Fuel - <input type="checkbox"/> OK              |
| 51. Fuel Boost - <input type="checkbox"/> On   | 42. Fuel - <input type="checkbox"/> OK              |
| 52. Fuel Boost - <input type="checkbox"/> On   | 43. Fuel - <input type="checkbox"/> OK              |
| 53. Fuel Boost - <input type="checkbox"/> On   | 44. Fuel - <input type="checkbox"/> OK              |
| 54. Fuel Boost - <input type="checkbox"/> On   | 45. Fuel - <input type="checkbox"/> OK              |
| 55. Fuel Boost - <input type="checkbox"/> On   | 46. Fuel - <input type="checkbox"/> OK              |
| 56. Fuel Boost - <input type="checkbox"/> On   | 47. Fuel - <input type="checkbox"/> OK              |
| 57. Fuel Boost - <input type="checkbox"/> On   | 48. Fuel - <input type="checkbox"/> OK              |
| 58. Fuel Boost - <input type="checkbox"/> On   | 49. Fuel - <input type="checkbox"/> OK              |
| 59. Fuel Boost - <input type="checkbox"/> On   | 50. Fuel - <input type="checkbox"/> OK              |
| 60. Fuel Boost - <input type="checkbox"/> On   | 51. Fuel - <input type="checkbox"/> OK              |
| 61. Fuel Boost - <input type="checkbox"/> On   | 52. Fuel - <input type="checkbox"/> OK              |
| 62. Fuel Boost - <input type="checkbox"/> On   | 53. Fuel - <input type="checkbox"/> OK              |
| 63. Fuel Boost - <input type="checkbox"/> On   | 54. Fuel - <input type="checkbox"/> OK              |
| 64. Fuel Boost - <input type="checkbox"/> On   | 55. Fuel - <input type="checkbox"/> OK              |
| 65. Fuel Boost - <input type="checkbox"/> On   | 56. Fuel - <input type="checkbox"/> OK              |
| 66. Fuel Boost - <input type="checkbox"/> On   | 57. Fuel - <input type="checkbox"/> OK              |
| 67. Fuel Boost - <input type="checkbox"/> On   | 58. Fuel - <input type="checkbox"/> OK              |
| 68. Fuel Boost - <input type="checkbox"/> On   | 59. Fuel - <input type="checkbox"/> OK              |
| 69. Fuel Boost - <input type="checkbox"/> On   | 60. Fuel - <input type="checkbox"/> OK              |
| 70. Fuel Boost - <input type="checkbox"/> On   | 61. Fuel - <input type="checkbox"/> OK              |
| 71. Fuel Boost - <input type="checkbox"/> On   | 62. Fuel - <input type="checkbox"/> OK              |
| 72. Fuel Boost - <input type="checkbox"/> On   | 63. Fuel - <input type="checkbox"/> OK              |
| 73. Fuel Boost - <input type="checkbox"/> On   | 64. Fuel - <input type="checkbox"/> OK              |
| 74. Fuel Boost - <input type="checkbox"/> On   | 65. Fuel - <input type="checkbox"/> OK              |
| 75. Fuel Boost - <input type="checkbox"/> On   | 66. Fuel - <input type="checkbox"/> OK              |
| 76. Fuel Boost - <input type="checkbox"/> On   | 67. Fuel - <input type="checkbox"/> OK              |
| 77. Fuel Boost - <input type="checkbox"/> On   | 68. Fuel - <input type="checkbox"/> OK              |
| 78. Fuel Boost - <input type="checkbox"/> On   | 69. Fuel - <input type="checkbox"/> OK              |
| 79. Fuel Boost - <input type="checkbox"/> On   | 70. Fuel - <input type="checkbox"/> OK              |
| 80. Fuel Boost - <input type="checkbox"/> On   | 71. Fuel - <input type="checkbox"/> OK              |
| 81. Fuel Boost - <input type="checkbox"/> On   | 72. Fuel - <input type="checkbox"/> OK              |
| 82. Fuel Boost - <input type="checkbox"/> On   | 73. Fuel - <input type="checkbox"/> OK              |
| 83. Fuel Boost - <input type="checkbox"/> On   | 74. Fuel - <input type="checkbox"/> OK              |
| 84. Fuel Boost - <input type="checkbox"/> On   | 75. Fuel - <input type="checkbox"/> OK              |
| 85. Fuel Boost - <input type="checkbox"/> On   | 76. Fuel - <input type="checkbox"/> OK              |
| 86. Fuel Boost - <input type="checkbox"/> On   | 77. Fuel - <input type="checkbox"/> OK              |
| 87. Fuel Boost - <input type="checkbox"/> On   | 78. Fuel - <input type="checkbox"/> OK              |
| 88. Fuel Boost - <input type="checkbox"/> On   | 79. Fuel - <input type="checkbox"/> OK              |
| 89. Fuel Boost - <input type="checkbox"/> On   | 80. Fuel - <input type="checkbox"/> OK              |
| 90. Fuel Boost - <input type="checkbox"/> On   | 81. Fuel - <input type="checkbox"/> OK              |
| 91. Fuel Boost - <input type="checkbox"/> On   | 82. Fuel - <input type="checkbox"/> OK              |
| 92. Fuel Boost - <input type="checkbox"/> On   | 83. Fuel - <input type="checkbox"/> OK              |
| 93. Fuel Boost - <input type="checkbox"/> On   | 84. Fuel - <input type="checkbox"/> OK              |
| 94. Fuel Boost - <input type="checkbox"/> On   | 85. Fuel - <input type="checkbox"/> OK              |
| 95. Fuel Boost - <input type="checkbox"/> On   | 86. Fuel - <input type="checkbox"/> OK              |
| 96. Fuel Boost - <input type="checkbox"/> On   | 87. Fuel - <input type="checkbox"/> OK              |
| 97. Fuel Boost - <input type="checkbox"/> On   | 88. Fuel - <input type="checkbox"/> OK              |
| 98. Fuel Boost - <input type="checkbox"/> On   | 89. Fuel - <input type="checkbox"/> OK              |
| 99. Fuel Boost - <input type="checkbox"/> On   | 90. Fuel - <input type="checkbox"/> OK              |
| 100. Fuel Boost - <input type="checkbox"/> On  | 91. Fuel - <input type="checkbox"/> OK              |



The Checklist: <https://www.newyorker.com/magazine/2007/12/10/the-checklist>

# Activity: Create your own checklist

- In pairs, think about dumb mistakes your “friend” made the last time they were coding.
  - Write your names on a piece of paper.
  - Write down two checklist items that would have caught those errors.
- Divide into teams: left and right sides of the classroom.
- Shout your ideas to Prof Begel, who will write them on the chalkboard.
  - Which team had the most unique/good entries in their list?
- By 5pm, upload a picture of your paper to Gradescope: October 5 Activity.

# Sample Low-Level Coding Checklist

(not complete)

- General
  - Are all changes relevant?
  - Do the classes and methods fulfill their purpose?
  - Are the messages and texts for the user correct?
- Classes
  - Are all assignments of attributes correct?
  - Are the classes implemented correctly?
- Arguments
  - Are the correct arguments used in all method calls?
- Recursion
  - Does recursion terminate properly?
- Methods
  - Do methods always return a valid value?
  - Do methods check parameters for validity (if needed)?
  - Are all parameters used?
  - Do methods have parameter and return types declared? Variables
  - Are all variables, counters, and accumulators initialized properly and, if necessary, re-initialized every time they are used?
  - Are all declared variables being used?
- If-Then Statements
  - Do the if-else statements fit the intended purpose?
  - Are all edge cases handled?
- Loops
  - Do the loops end under all possible conditions?
  - Are the break and continue statements used properly?
- Errors
  - Are exceptions handled correctly?
- Final Check
  - Are all changes consistent with one another?

# Formal Inspections

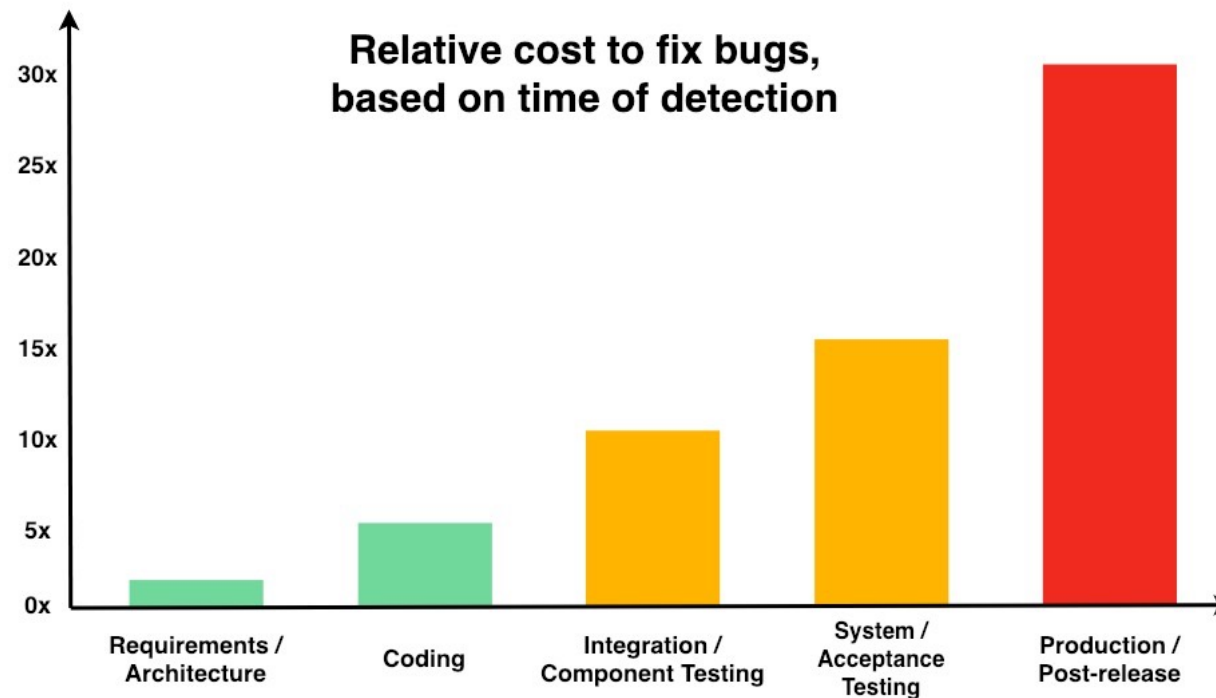
- Idea popularized in 70s at IBM
- Broadly adopted in 80s, much research
  - Sometimes replaced component testing
- Group of developers meets to formally review code or other artifacts
- Most effective approach to find bugs
  - Typically, 60-90% of bugs found with inspections
- Expensive and labor-intensive

# □ Inspection Team and Roles

- Typically, 4-5 people (min 3)
- Author
- □ Inspector(s)
  - Find faults and broader issues
- Reader
  - Presents the code or document at inspection meeting
- Scribe
  - Records results
- Moderator
  - Manages process, facilitates, reports

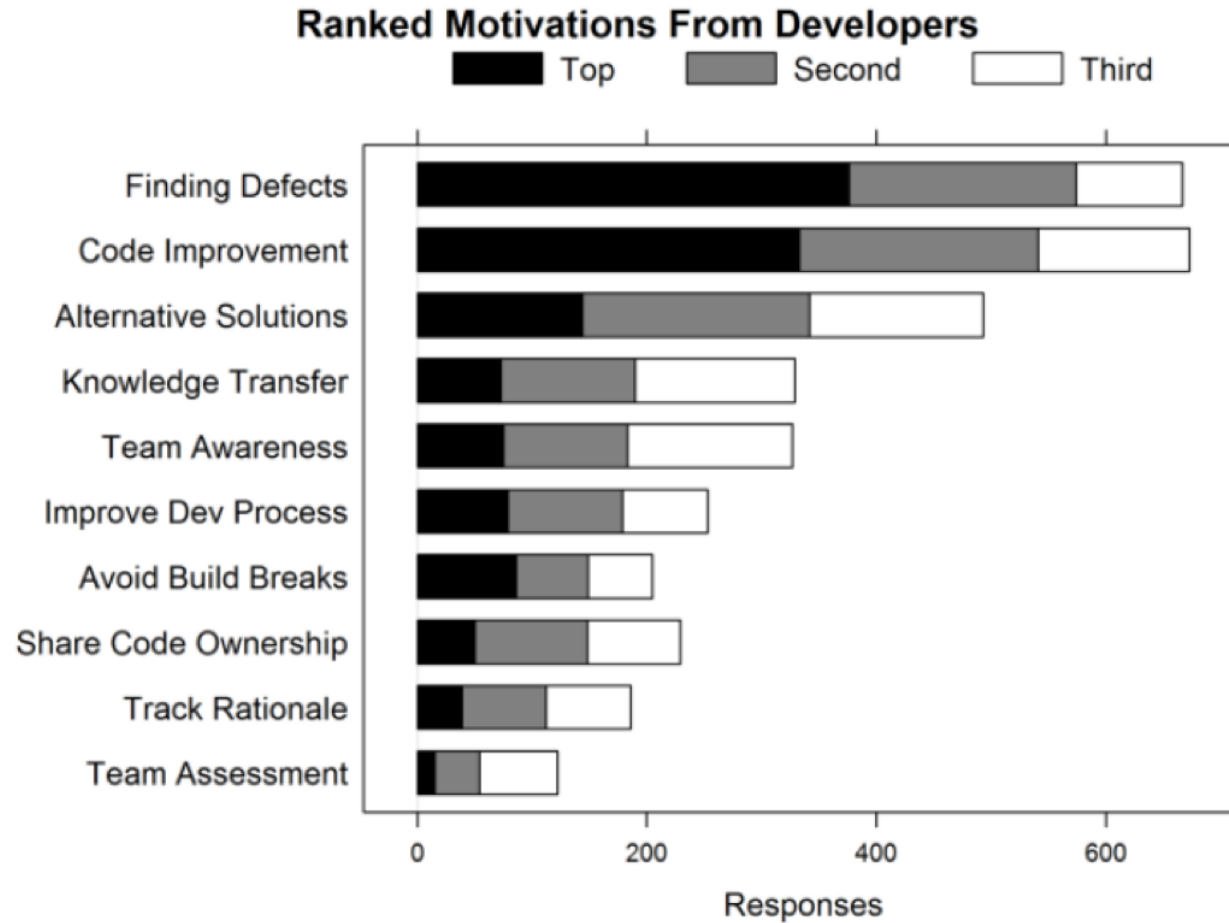
# Motivation

- Linus's Law: "Given enough eyeballs, all bugs are shallow."
  - - The Cathedral and the Bazaar, Eric Raymond



# Expectations and Outcomes

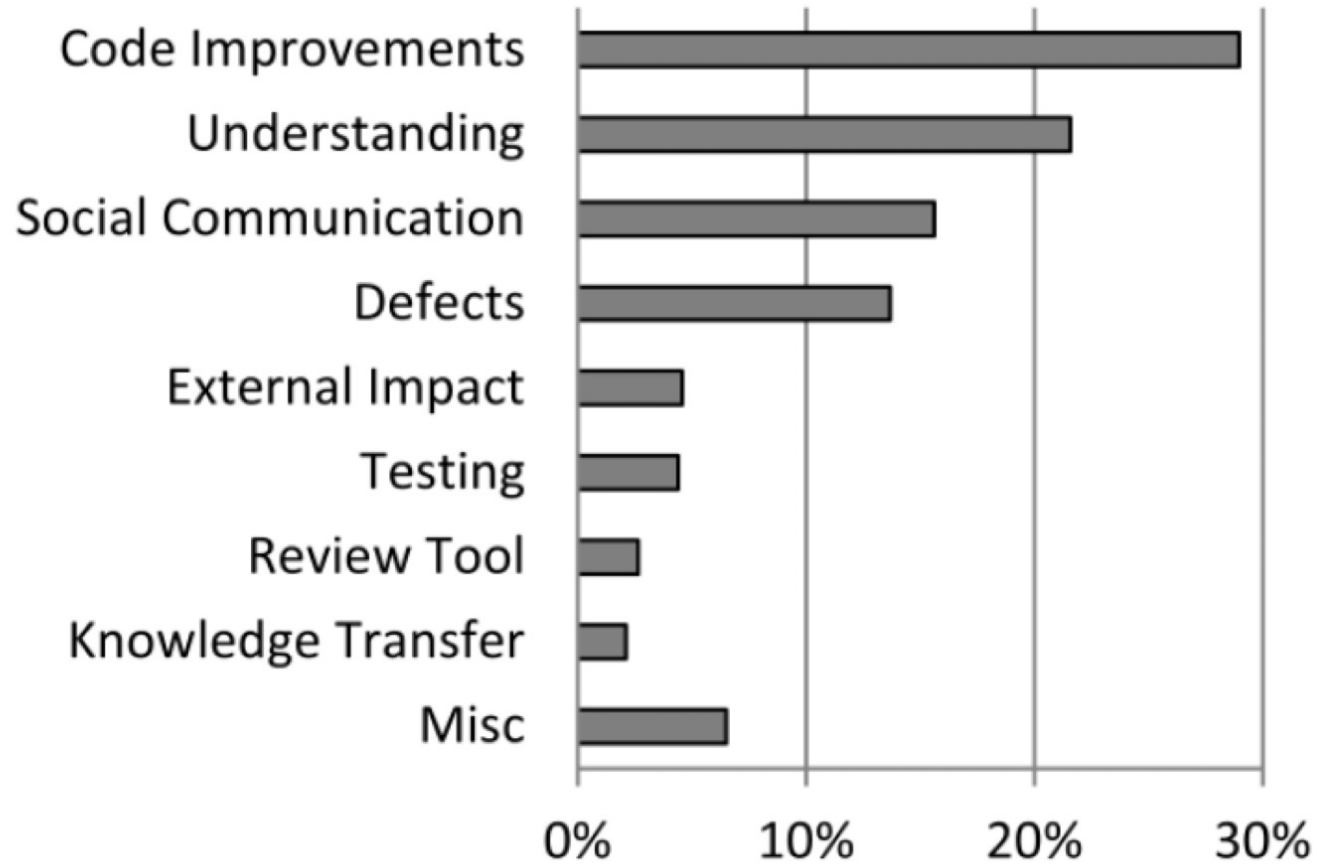
# Code Review at Microsoft



Bacchelli, Alberto and Christian Bird. "Expectations, outcomes, and challenges of modern code review." Proceedings of the 2013 International Conference on Software Engineering. IEEE Press, 2013.



# Outcomes (Analyzing Reviews)



# Mismatch of Expectations and Outcomes

- Low quality of code reviews
  - Reviewers look for easy errors, as formatting issues
  - Miss serious errors
- Understanding is the main challenge
  - Understanding the reason for a change
  - Understanding the code and its context
  - Feedback channels to ask questions often needed
- No quality assurance on the outcome

# Code Review at Google

- Introduced to “force developers to write code that other developers could understand”
- Three benefits:
  - checking the consistency of style and design
  - ensuring adequate tests
  - improving security by making sure no single developer could commit arbitrary code without oversight

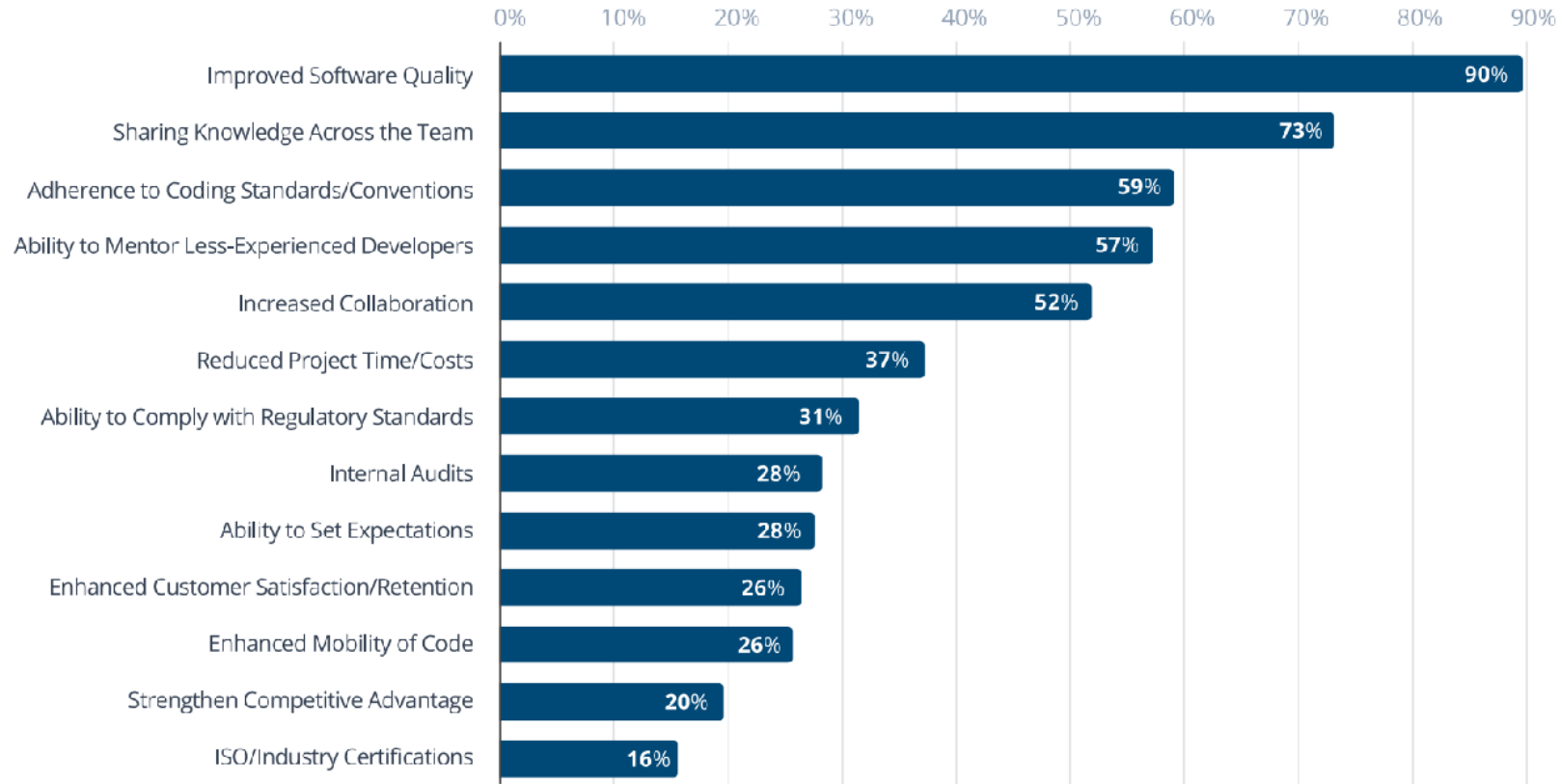
Caitlin Sadowski, Emma Söderberg, Luke Church, Michal Sipko, and Alberto Bacchelli. 2018. Modern Code Review: A Case Study at Google. International Conference on Software Engineering

# Reviewing Relationships



# The State of Code Review survey

What do you believe are the most important benefits of code review?



n = 1129

# Code Review

- Start with the “big ideas”
- Automate the little things
- Focus on understanding
- Remember a person wrote the code
- Don't overwhelm the person with feedback

# Don't forget that coders are people with feelings

- A coder's self-worth is in their artifacts
- C can avoid embarrassment
- Identify defects, not alternatives; do not criticize coder
  - “you didn't initialize variable a” -> “I don't see where variable a is initialized”
- Avoid defending code; avoid discussions of solutions/alternatives
- Reviewers should not “show off” that they are better/smarter
- Avoid style discussions if there are no guidelines
- The coder gets to decide how to resolve fault