

# Studying the Clarity of Code Review Comments

\* Required

## Introduction

Dear Participant,

We are researchers from X University, we are now conducting a research project to investigate what are clear code review comments.

**Background:** Code review is the manual inspection of source code by additional professionals other than the author of the submitted code. It is a crucial activity for the software quality assurance of both open source and commercial software systems. When reviewing the code changes, the reviewer can provide comments for any part in the submitted code to indicate whether the code looks good, need to be revised, or the reviewer has any questions. Then the author of code changes can reply or act accordingly. The figure below shows an example of code review (The author and date are hidden for privacy).

**Task:** A well-written code review comment should clearly convey the ideas of the reviewer. However, it is challenging to systematically evaluate what constitutes the "clarity" of code review comments. Thus, this survey aims to investigate the characteristics of clear code review comments and the potential directions on improving code review comment generation techniques, through a series of multiple-choice and open-ended questions.

The survey takes around 5 minutes. We will not reveal your identity in any form of the research outcomes. All of the information provided in the survey will be used for research purposes only.



## Part I: Information of Participant

1

**What is your country/region of residence? \***

2

**What is your current occupation? \***

- ☐ Industrial or freelance professional
- ☐ Academic or industrial researcher
- ☐ Graduate or undergraduate student
- ☐ Other

3

**Which of the following best describes your primary job role? \***

- ☐ Development
- ☐ Testing
- ☐ Software Project Management
- ☐ Research
- ☐ Other

4

**How many years of experience do you have with your primary job role? \***

- ☐ ≤1 year
- ☐ 2 years
- ☐ 3 years
- ☐ 4 years
- ☐ 5 years
- ☐ 6 years
- ☐ 7 years
- ☐ 8 years
- ☐ 9 years
- ☐ ≥10 years
- ☐ Other

5

**Which of the following programming languages are you familiar with? \***

- ☐ Python
- ☐ C
- ☐ C++
- ☐ C#
- ☐ Java
- ☐ JavaScript
- ☐ Go
- ☐ PHP
- ☐ Rust
- ☐ Ruby
- ☐ Other

**Do you have code review experience in open-source projects and/or industrial projects? \***

☐ Yes

☐ No

## Part II: Aspects of the Clarity of Code Review Comments

Based on our literature review, we derive three aspects that are related to the clarity of code review comments: **Relevance**, **Informativeness**, and **Expression**. Please answer the questions related to each aspect and let us know your ideas on other potential aspects.

7

### A1 - Relevance: If the code review comment is relevant to the code change.

Below, we list a complying example and a violating example to illustrate this aspect:

#### [Complying Example]

(code change)

```
+  
+void wlr_egl_destroy_surface(struct wlr_egl *egl, EGLSurface surface) {  
+    eglDestroySurface(egl->display, surface);  
+}
```

(code review comment)

Could you add a NULL check on surface, and then remove the checks from the DRM backend?

#### [Violating Example]

(code change)

```
- pony_assert(token != NULL);  
+ pony_ASSERT_TOKEN_NOT_NULL(token);
```

(code review comment)

I'm not in favor of this change.

**Question: To what extent do you think that the aspect of "Relevance" is important to the clarity of code review comments? \***

- ☐ Very important
- ☐ Somewhat important
- ☐ Neutral
- ☐ Somewhat unimportant
- ☐ Very unimportant

8

**If you are evaluating a code review comment, how will you evaluate its relevance to the code change?** (e.g., what factors and detailed information you may focus on)

9

**Could you please explain the reason of your choice?**

**A2 - Informativeness:** If the code review comment provides sufficient information.

Below, we list a complying example and a violating example to illustrate this aspect:

**[Complying Example]**

```
(code change)
+ def clear(self, key):
+     """
+     Removes all items with the specified key, and does not raise an
+     exception if the key does not exist.
+     """
+     if key in self:
+         del self[key]
```

(code review comment)

I think we should stick to the standard API and use `.pop(key, None)` instead of inventing our own `clear()`

**[Violating Example]**

```
(code change)
- self.invalidate_cache()
  self.resync()
+ self.invalidate_cache()
```

(code review comment)

Why?

**Question: To what extent do you think that the aspect of "Informativeness" is important to the clarity of code review comments? \***

- ☐ Very important
- ☐ Somewhat important
- ☐ Neutral
- ☐ Somewhat unimportant
- ☐ Very unimportant

**If you are evaluating a code review comment, how will you evaluate its informativeness?** (e.g., what factors and detailed information you may focus on)

**Could you please explain the reason of your choice?**

**A3 - Expression:** If the code review comment is readable, easy to understand, and friendly.

Below, we list a complying example and a violating example to illustrate this aspect:

**[Complying Example]**

(code change)

```
- return x if isinstance(x, bytes) and not six.PY2 else x.encode("hex")
+ return bytes_hex(x).decode()
```

(code review comment)

What is the expected difference between `to\_hex()` and `bytes\_hex()`?

**[Violating Example]**

(code change)

```
+PGSQL_MAX_INTEGER_VALUE = 9223372036854775807
```

(code review comment)

```
`2**64 / 2`
```

**Question: To what extent do you think that the aspect of "Expression" is important to the clarity of code review comments? \***

- ☐ Very important
- ☐ Somewhat important
- ☐ Neutral
- ☐ Somewhat unimportant
- ☐ Very unimportant

**If you are evaluating a code review comment, how will you evaluate its expression?** (e.g., what factors and detailed information you may focus on)

**Could you please explain the reason of your choice?**

**Do you have some ideas about other important aspects that contribute to the clarity of code review comments?**

### Part III: Questions on Automated Code Review Comment Generation

17

**Have you used some tools to generate code review comments? \***

☐ Yes

☐ No

18

**Could you please specify what tools you have used and your overall feedback on them (e.g., how is the clarity of the generated results)?**

19

**How would you rate the importance of the clarity of generated code review comments to the user experience of such tools? \***

☆ ☆ ☆ ☆ ☆

20

**Could you please explain the reason of such rating?**

21

**If there is tool which can generate code review comments with good clarity, would you like to use it to assist you in writing code review comments? \***

☐ Yes

☐ No

☐ Not sure

22

**Could you please explain the reason of the answer above?**



**Open-ended Question: Do you have other ideas/considerations/concerns about code review and its generation? Any ideas or suggestions are highly welcomed!**

---

This content is neither created nor endorsed by Microsoft. The data you submit will be sent to the form owner.



Microsoft Forms