



How to apply a code  
code review

Tips for better  
coexistence

Mauricio R Duarte



Mauricio R Duarte



@mauriciord



@mauriciord



Front End developer

## Before we start

"This presentation just goes through a possible exercise that can help in code review processes, based on references that will be passed at the end of the presentation. Therefore, it is no rule, just an exchange of experiences."

# Overview

- Code review as Mentoring
- Code review principles
- What to look for in code review
- Examining and navigating a code review
- Speed of a code review
- How to write code review comments

## Code review as Mentoring

Code review is a good time between exchanging knowledge. It is where developers something new about a language, a framework, or general software design principles. Sharing knowledge is part of improving the code health of a system over time. Always try to leave comments helping or congratulating the other team members work, even because bad comments generate bad environments.



# Code review principles

- Technical facts and data replace opinions and personal preferences.
- The style guide is the absolute authority. If there is no previous style, accept the author's.
- Aspects of software design are almost never a pure style issue or just a personal preference.

## What to look for in code review

- The code is well-designed.
- Code has appropriate unit tests
- Tests are well-designed.
- The code conforms to our style guides.
- Any UI changes are sensible and look good.
- The developer isn't implementing things they might need in the future but don't know they need now.
- The developer used clear names for everything.
- Any parallel programming is done safely.
- The code isn't more complex than it needs to be.
- Code is appropriately documented.
- The functionality is good for the users of the code.
- Comments are clear and useful, and mostly explain why instead of what.

## Examining and navigating a code review

1. Full view of the Pull Request.
2. Find the file or files that are the "main" part or ask the developer what you should look at first
3. Look the rest of the sequence or try to read the tests before read the main code



Speed of a code review



## Speed of a code review

### Why Should Code Reviews Be Fast?

Instead of giving speed to an individual developer, we should give speed to team, to build a product with speed

### How Fast Should Code Reviews Be?

If you are not in the middle of a focused task, **you should do a code review shortly after it comes in.**

### Speed vs. Interruption

Interrupting yourself while coding is actually more expensive to the team than making another developer wait a bit for a code review.

## How to write code review comments



## How to write code review comments

### Courtesy

Be sure that you are always making comments about the code and never making comments about the developer

Good

+ 😊 ...

Talvez daria pra eliminar esse comentário se extrair esse método, criar a variável do elemento por referência de memória e atribuir os parâmetros no mesmo:

```
function (el) {  
  const theElement = el;  
  theElement.expando = {};  
}
```

## How to write code review comments

### Explain Why

Helps the developer understand why you are making your comment.

### Giving Guidance

In general, you should strike an appropriate balance between pointing out problems and providing direct guidance. You are not required to do a detailed design of a solution or write code for the developer.

### Accepting Explanations

Ask a developer to explain a piece of code that you don't understand.

**Explanations written only in the code review tool are not helpful to future code readers.**

## Whats next?

- [Learn In Public](#)
- [The Product-Minded Software Engineer](#)

## References

- [Google's Engineering Practices documentation](#)
- [Pull requests and code review](#)
- [Palantir: Code Review Best Practices](#)

**Thank you**

We are hiring :]

**Want to contribute?**

[Code review presentation repo](#)