

Revisão de Código



Agenda

1

O que é Revisão de Código

2

Por que fazer Revisão de Código?

3

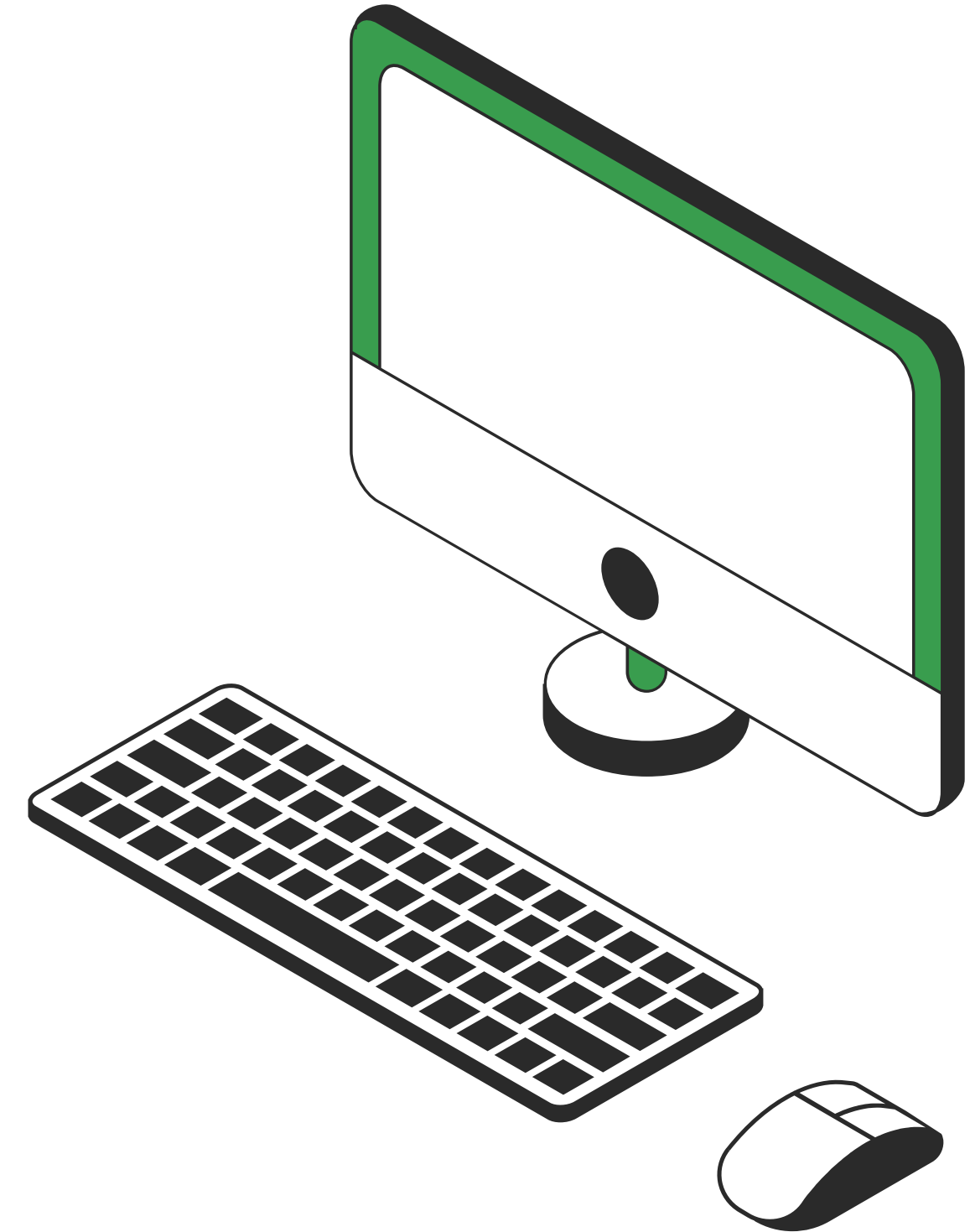
Que aspectos posso considerar numa revisão?

4

[Time] Acordos de revisão e Merge Request Template

O que é Revisão de Código

É o processo em que o código desenvolvido é submetido a análise de outras pessoas que o revisam, podendo identificar possíveis problemas/bugs e oportunidades de melhorias, oferecendo feedbacks para garantir a qualidade antes de integrá-lo à base.



Por que fazer Revisão de Código?

1

Maior rapidez em identificar possíveis bugs.

2

Mantemos nosso código alinhado com acordos de qualidade.

3

Compartilhar contexto e trocar experiências.



**Como revisar o
código?**

Como revisar o código?

Algumas dicas :)

O que devo informar ao pedir um review?

É importante que quem pede a revisão de código, forneça algumas informações aos revisores como:

- Descrição da alteração e um overview do motivo.
- Qual o escopo da mudança.
- Trechos do código que precisam de mais atenção, mais críticas.

Colabore e peça colaboração

Mesmo que você não tenha sido marcado como revisor e achar que pode contribuir na revisão do código de alguém, não tenha medo!

Code Review é um exercício de empatia.

Faça comentários e compartilhe ideias levando em consideração a pessoa que vai receber. Lembre que o objetivo não é que ela aprenda como você faria, então a lógica do colega deve prevalecer.

Revisão de Código

Podemos estabelecer alguns aspectos que devemos dar atenção no processo de revisão:

Operacional

Qualidade

Legibilidade
do Código

Manutenibilidade

Design do
Código

Revisão de Código – Operacional

Se o novo código atende os requisitos do ponto de vista operacional:

1

As saídas são os valores esperados para o conjunto de entrada?

2

Existe alguma validação para os tipos de entrada?

3

Como a solução iria performar em uma escala maior?

4

Essa solução pode desencadear algum problema de consumo de memória?

Revisão de Código – Legibilidade

1

Estamos seguindo o padrão do idioma escolhido?

2

Seguimos algum padrão para nomes (ex: camel case, underscore)?

3

O nome do método/variável/classe transmite a ideia do que ele realmente faz?

4

Existe README? é útil?

5

A leitura do código é fácil?

Revisão de Código – Manutenibilidade

Seria difícil outra pessoa desenvolvedora dar manutenção no novo código?

1

A solução é simples?
Não poderia ser
menos complexa?

2

Existe algum método,
função, variável
inútil? Podemos
remover?

3

Existe excesso de
comentários no
código?

4

Alguma duplicação
no código? Algum
método pode ser
extraído?

5

Estamos quebrando
regra de
responsabilidade
única?

6

Existem mensagens
de log que apoiem a
investigação?
Podemos monitorar
o novo código?

Revisão de Código – Qualidade

1

Foram escritos testes unitários?

2

A cobertura de testes desse código é aceitável?

3

Precisa ser desenvolvido algum teste funcional ou integrado?

4

As possíveis entradas dessa solução foram testadas?

5

Deixamos algum débito no código novo?

6

Atende a todos os critérios de qualidade? O Sonar é uma ferramenta de apoio desses requisitos.

Revisão de Código – Design e Arquitetura

1

Nomes: Os nomes das classes/funções/variáveis estão ok?

2

Organização: Os arquivos novos foram distribuídos nos diretórios corretos?

3

A separação dos conceitos estão consistentes?

4

O código está acoplado? Existe algo que não pertence a classe/método em análise?

5

Estamos lidando corretamente com as exceções?

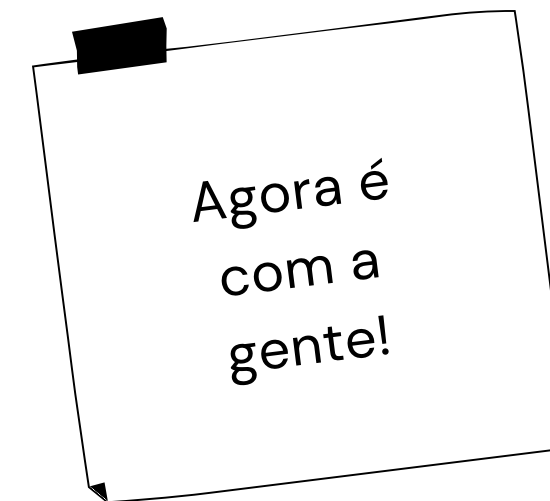
6

Aplica os princípios SOLID? A responsabilidade do componente está clara?

Vamos começar!

Acordos de revisão e Merge Request Template

[Acesso os acordos aqui](#)



Obrigada!

